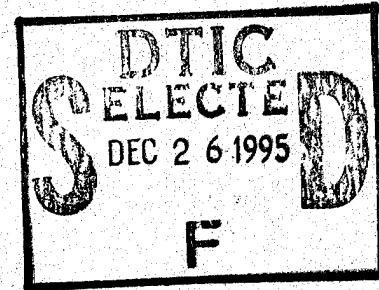
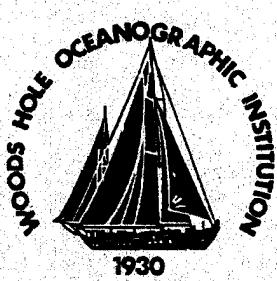


**Woods Hole  
Oceanographic  
Institution**



---

**Radio LAN Acquisition Module (RLAM),  
Recent Developments for High Resolution Data Collection  
Systems as Implemented for the ONR Sea Ice  
Mechanics Experiment, Spring 1994**

by

K. von der Heydt, C.F. Eck

*\*Original contains color plates: All DTIC reproductions will be in black and white\**

May 1995

**Technical Report**

Funding was provided by the Office of Naval Research under  
Contract No. N00014-91-J-1296.

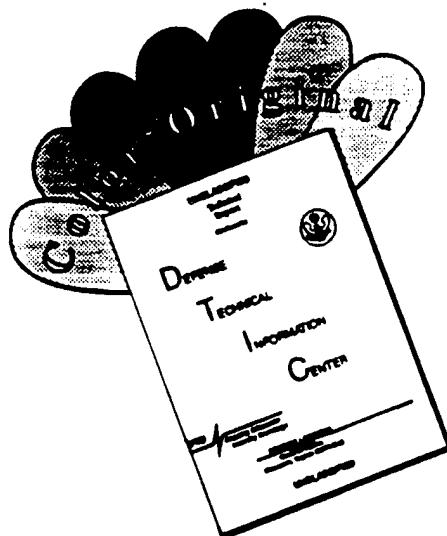
Approved for public release; distribution unlimited.

---

19951219 004

DTIC QUALITY INSPECTED 1

# DISCLAIMER NOTICE



THIS DOCUMENT IS BEST  
QUALITY AVAILABLE. THE COPY  
FURNISHED TO DTIC CONTAINED  
A SIGNIFICANT NUMBER OF  
COLOR PAGES WHICH DO NOT  
REPRODUCE LEGIBLY ON BLACK  
AND WHITE MICROFICHE.

**WHOI-95-09**

Radio LAN Acquisition Module (RLAM),  
Recent Developments for High Resolution Data Collection Systems as Implemented  
for the ONR Sea Ice Mechanics Experiment, Spring 1994

by

**K. von der Heydt, C.F. Eck**

Woods Hole Oceanographic Institution  
Woods Hole, Massachusetts 02543

**May 1995**

**Technical Report**

Accession For	
NTIS	CRA&I <input checked="" type="checkbox"/>
DTIC	TAB <input type="checkbox"/>
Unannounced <input type="checkbox"/>	
Justification	
By _____	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

Funding was provided by the Office of Naval Research under  
Contract No. N00014-91-J-1296.

Reproduction in whole or in part is permitted for any purpose of the United States  
Government. This report should be cited as Woods Hole Oceanog. Inst. Tech. Rept.,  
WHOI-95-09.

Approved for public release; distribution unlimited.

**Approved for Distribution:**

George V. Frisk

George V. Frisk, Chair

Department of Applied Ocean Physics and Engineering

# Radio LAN Acquisition Module (RLAM), Recent Developments for High Resolution Data Collection Systems as Implemented for the ONR Sea Ice Mechanics Experiment, Spring 1994

K. von der Heydt, C. F. Eck  
Woods Hole Oceanographic Institution, Woods Hole, MA

May 8, 1995

## Abstract

During a recent experiment (April 1994), for the ONR Sea Ice Mechanics Initiative (SIMI), a portable data acquisition system was assembled that included 2 new developments. The first consists of a board, designed for the ISA PC bus incorporating 8 - 24 bit sigma-delta analog-to-digital converter (ADC) channels with 20 bit rms dynamic range. Among the features are programmable bandwidth to 1500 Hz, low power dissipation, digital anti-alias filtering, and a "floating point" mode resulting in a 16 bit word. Secondly, since the telemetry of data at continuous rates in excess of 100Kbytes/s was required, hardware & software was developed to use a wireless LAN to network 3 sites up to 5km distant from the data recording system. Details of the system along with test data are described.

## Contents

<b>1 Scientific Motivation</b>	<b>3</b>
<b>2 RLAM Technical Requirements</b>	<b>3</b>
<b>3 The RLAM's</b>	<b>5</b>
3.1 Sensors . . . . .	5
3.2 High Resolution 8 Channel ADC Board . . . . .	6
3.2.1 24 Bit Mode . . . . .	8
3.2.2 Pseudo Floating Point Mode . . . . .	8
3.2.3 Fixed Point Mode . . . . .	8
3.2.4 Data Storage . . . . .	8
3.2.5 Sequencer . . . . .	9
3.2.6 Use of Multiple AD24's . . . . .	9
3.2.7 Sample Count Interrupt . . . . .	9
3.2.8 Power . . . . .	9
3.2.9 AD24 Performance . . . . .	11
3.3 Telemetry . . . . .	13
3.4 Receiving & Storage System . . . . .	14
3.5 Power . . . . .	17
3.6 Software . . . . .	18
<b>4 Acknowledgments</b>	<b>21</b>
<b>5 List of Figures and Tables</b>	<b>22</b>
<b>A Appendix, AD24 Functional Details</b>	<b>23</b>
A.1 Operation . . . . .	23
A.2 Sequencer Functions . . . . .	24
<b>B Appendix, Battery Test Data</b>	<b>27</b>
<b>C Appendix, AD24 Configuration Code</b>	<b>28</b>

## 1 Scientific Motivation

The SIMI experiment in the spring of 1994 brought researchers together from the ice mechanics, oceanographic, and acoustic communities to observe and gather data related to sea ice fracturing mechanisms. The overall objective of this program was a better understanding of the mechanical properties of ice, and their relation to environmental forcing.

The focus of the MIT/WHOI component of the SIMI-94 experiment was the use of acoustic detection methods to localize ice events followed by measurements in the immediate vicinity of detected event sites to gather data related to the elastic behavior of sea ice. The intent of our work is to develop seismo-acoustic methods for modeling the mechanical properties of ice using the data acquired at ice crack event sites.

The technical developments described here were motivated by the need to gather data at event sites with quickly deployed measurement systems. Three axis geophones were the primary sensor.

During the experimental period, a horizontal array of 32 hydrophones with an aperture of about 500m and a time domain beamformer running on a UNIX workstation was used as a surveillance tool to detect and localize ice events in realtime. The results allowed us to monitor and inform other participants of the general level and location of ice activity within a radius of about 3km of the array. The objective was to chose sites for near field geophone deployments based on a combination of surveillance array detections and direct observations.

Shortly after the ice camp was established, local ice activity became so intense that most personnel were temporarily evacuated from the camp. This would have been the ideal time to have our portable systems deployed but neither we nor our equipment were ready. As we progressed with equipment preparations, the ice activity diminished. Only during the last week of the experiment period was lead formation and ridging apparent. We were able, during this period to collect about 120GB of data using the system described in this report.

Figure 1 is a mock aerial view of the Spring SIMI-94 camp. The violet squares indicate the relative locations of Radio Lan Acquisition Module (RLAM) deployments. The bluish coloration is indicative of actual ice floe boundaries. Approximate distances between the RLAM's and the VLA site are shown. The receiving antenna and data logging computer were installed at the larger green hut shown near the VLA.

## 2 RLAM Technical Requirements

The term "near field" refers to signals from the elastic response of ice to cracking events on scales of millimeters (thermally generated) to many meters (ridging and rafting processes) before they are garbled by scattering effects that occur at floe and ice crack boundaries. Therefore we had to place geophones (velocity measuring sensors), in the immediate vicinity of crack sites. Normally these are the areas where we would avoid deploying sensors because of the likelihood of wire breaks and sensor losses in the shifting ice.

The RLAM was designed to permit data collection under these conditions. A distributed system with short cabling could be deployed piecemeal and was less likely to suffer total loss of data acquisition capability in difficult ice conditions.

Personnel are a source of acoustical signal contamination so it was desirable to have these systems autonomous once deployed. Fast deployments were critical to minimize the time between identification of an active area and actually being in position to take data. These conditions required that systems either record data internally for periods of days or, more desirably, telemeter large quantities of data to a manned, centrally located site at some distance.

Any system put into the Arctic environment must withstand the extremes of low temperature (to  $-50^{\circ}$  C), wind, brittleness due to the cold and general rough handling as well as sled and helicopter transportation.

Radio linked autonomous systems minimize the amount of wire needed to connect sensors. They can be easily recovered for redeployment elsewhere, can make data available in near realtime to establish its quality and minimize the need for personnel at remote sites. Key to the feasibility of this project was a commercially available radio Ethernet adapter that supported the standard TCP/IP protocol. Such a system had been configured for a previous experiment to continuously transmit data from a buoy to a nearby ship. [1]

The original "wish list" for a near-field system was:

- small vertical line array of hydrophones....possibly 8 ch
- a quantity of 3-axis geophones, 50-2kHz bandwidth
- high dynamic range... $\geq 16$  bits
- programmable sample rate with band limiting
- fast deployment/recovery, via helicopter or snowmobile
- modest power requirements
- onsite recording system that could be installed/recovered quickly with the sensors and operated autonomously in a preconfigured hut

Previous work had shown that data could be telemetered in real-time directly to the main camp using a radio LAN. However there is a significant reduction in the net data bandwidth originally posed. During an earlier part of the Sea Ice Mechanics program (November 1993) we had the opportunity to test the LAN hardware under realistic conditions with various antennas. These tests suggested that the 5km range and at least 100K Byte/sec aggregate data rates were possible. Unfortunately, this rate was a fraction of what we needed. However the attractiveness of reliable radio telemetry was compelling and compromises were made. As a result, we prepared the following for use during the April 1994 SIMI field work:

- Four 16 channel battery powered systems, each with 5 3-axis geophones and 1 hydrophone. configured as shown in Figure 2
- A selection of programmable sampling rates with a maximum of 3906 Hz, (1500 Hz BW)
- Two radio links, one for data telemetry and one for a power cycling and timebase sync
- 10 slot ISA chassis containing a DX 486 single board computer (SBC), NCR WaveLAN Ethernet adapter, timing and power control board, 2 - 8 ch 24 bit analog-to-digital converter (ADC) boards and a DC/DC converter.
- 2 kWh battery pack, for 1 week Arctic operation
- Well insulated orange polypropylene packing case with custom through-fitting for sensor cables and a snowproof nylon "sock"for cable connections
- PC based receiving system with Exabyte drive for data storage and the ability to download code to the remote sites.

### 3 The RLAM's

Each RLAM system consisted of the components shown in Figure 3 and was deployed in an orange polypropylene packing case (to aid visibility on pack ice), insulated and sized to snugly hold the RLAM electronics chassis and a pair of battery packs. The insulation used was urethane foam ( $R=6$  per inch), 4" in the top, 2" in the sides, and 1" in the bottom. Space was allowed for 2 identical alkaline battery packs which were situated above (actually rested on) the electronics chassis to be warmed by the approximately 26 watts of power dissipated when the system was running.

A 4" diameter hole in the side of the RLAM case held a threaded plastic fitting that could be sealed during shipment and field transportation. It allowed easy access to cables for sensor and antenna connections. Figure 4 is a photograph showing an RLAM and its mast with 2 antennas mounted.

Each RLAM was configured to acquire data from 16 channels at a common selected rate. During episodes using more than 1 RLAM, all were configured from the LAN link to run at the same rate. Fifteen channels were devoted to 5 - 3 axis geophones. An omni directional broadband hydrophone was connected to the 16th input on each RLAM. Inexpensive multi-pin automotive type connectors were used to connect sensor cables.

The RLAM's were computationally powered by a DX486 based single board computer (SBC) running on a passive backplane in a 10 slot chassis. Though the arithmetic capability of a DX processor was really unnecessary during data acquisition operations, it was convenient for a development/checkout mode so data could be manipulated with MATLAB.

The SBC was a TEKNOR AT4 half size board that incorporates an entire PC based on a choice of SX/DX processors clocked at 25 to 66 MHz. The AT4 incorporates 4 - 30 pin DRAM slots (we used 16M of DRAM), 1M of FLASH EPROM memory that can be configured as the "boot" disk, 128K of static RAM, IDE & floppy interfaces, serial & parallel ports, realtime clock and "deadman". It's only drawback was that at about \$1295 with an 486SX processor it was expensive.

An AMD DX-40 processor running at 33 MHZ was used. It was found to dissipate about the same power as an Intel 486SX running at 25 MHz. A VEMALINE tower heatsink was attached to the processor and was held against the side of the chassis as a heat dissipation measure in the enclosed insulated RLAM case. Though lower power "SL" type processors were available, no other SBC at the time provided the desired mix of features and easy use of an onboard FLASH memory boot disk along with the lower power processor.

A 10 slot chassis was more than adequate but the additional space left enough room for a copper shield separating the 2 ADC boards from the SBC, WaveLAN, and timebase interface. Subsequent testing has indicated that the copper shield plate has little if any effect on data quality. During development and testing another slot was taken by a video monitor interface.

#### 3.1 Sensors

Four 16 channel systems were assembled, each configured for 5 3-axis geophones and 1 hydrophone. The geophone elements were Western Geophysical 4.5 Hz (M/N LRS1011 inside an LRS1033 case) with useful bandwidth from resonance to at least 1 kHz and sensitivity of 642mv/in/sec at 70% damping. The coil resistance of these geophones is  $420 \Omega$  and they cost about \$30 in small quantities. Each system included one omnidirectional hydrophone, designed and manufactured at WHOI, with a low frequency -3dB point at 1 Hz and a sensitivity of -160 dBv re  $1 \mu\text{P}$ . The hydrophone was deployed at a depth of 60m, common to all hydrophones in our horizontal event localization array as depicted by the red circles in Figure 1.

Geophone manufacturers describe a region of narrowband "spurious response" that is typically above the frequency region of land-seismic interest and therefore ignored, but potentially of concern to us. Western Geophysical tells us that these units can exhibit this seemingly resonant condition in the region of 300 Hz. It seems difficult to get information quantifying this response and the reasons for it, beyond the physical construction of the moving mass-within- a-coil sensing element itself. We have not observed such a peaked narrow band signal in real data.

Each 3-axis geophone was a single polyethylene package about 5" long by about 2" square with a single cable consisting of 3 individually foil shielded twisted pair emerging from a water-tight gland at one end. The attached cable was 200 feet long and, though a little stiff and unwieldy in the cold, eliminated the need for a connector at the sensor end. Inexpensive molded automotive style, polarized, 6 pin connectors were used to connect geophones to the RLAM's. Their only drawback is that they stiffen sufficiently in the cold to make connection/disconnection difficult. During RLAM deployments, we usually had a small "salamander" style heater and a small generator available that eliminated this problem.

The geophones contain 2 identical horizontal sensors and one vertical sensor mounted orthogonally to each other. A damping resistor of 3.48K shunted the sensors at their solder connections. Power is not required for the geophones. An alcohol filled bubble level is integrally mounted on top of the geophone case as an aid to leveling during deployment.

Geophone installation required a small area to be cleared of snow so that the sensor could be attached directly to the ice with fresh water mixed with snow, otherwise known as "arctic concrete", to form a fast freezing slurry. As a 3-axis unit was placed on the ice, some of this slurry was poured around while it was held level according to the bubble. Mounting spikes, normally used for land seismic installations, were not used. A small fixture, placed over the geophone was used to align all units in the same (nominally north) direction. After deploying geophones, their relative positions were surveyed using a sextant to measure angles between another sensor and the centrally located RLAM antenna pole.

The single hydrophone was deployed through a hole drilled with a hand auger or simply put into the water where a lead in the ice had recently opened. Unlike the phones deployed as part of the horizontal array, which were in air filled tubes, the RLAM hydrophones usually had to be recovered by drilling a second hole near the first and "snatching" the wire beneath the ice.

### 3.2 High Resolution 8 Channel ADC Board

A new development for this project was an 8 channel Industry Standard Architecture (ISA) board ADC based on a recently released 24 bit Sigma-Delta architecture chip set from Crystal Semiconductor Corp., (CSC). Our intention was to use this board with standard, inexpensive multiple slot passive ISA backplanes, for both the SIMI work and future projects. Therefore, the board was designed, as much as practical, as a general purpose, low frequency, expandable ADC for use as part of any 16 bit ISA system. An obvious concern with such dynamic range was noise from processors, logic and power supplies as well as crosstalk from other AD24 cards. Care was taken during layout to minimize onboard contamination of signals, resulting in a 6 layer board with multiple power planes largely populated with surface mount devices. On-board linear regulators for the analog circuitry and the S/D modulators were necessary to prevent power supply noise as a function of high frequency digital logic surges from contaminating the data.

Though data collection efforts were ultimately successful, one noise problem was the source of much grief during the SIMI-94 experiment until we improved the shielding of analog signal cables and subsequently added small value bypassing capacitors at the differential amplifier inputs. A discussion of this problem is included in a later section.

Each RLAM system contained two of these eight channel ADC boards in a 10 slot chassis to process 16 data channels. The Sigma-Delta converter (SDC) topology requires a converter per channel approach, assuring simultaneous sampling of all channels. A characteristic of the SDC approach is a highly oversampling, (typically 1 bit), front end which band shifts the quantization noise up in frequency. The filter/decimation function (1 device of the 2-chip set), employs an FIR digital filter with a bandwidth of about 40% of the output sample rate.

The use of a 24 bit system was motivated by the need for a large measurement range of less complexity than a system with multiple discrete gain settings and the means to automatically select the optimal gain on the fly. In fact Crystal demonstrates their chip set to be about a 20 bit system at 1 kHz output rate. In our implementation we realize a remarkably similar rms dynamic range at that rate, despite the proximity of ISA bus activity. The board was designed to operate in 1 of 3 modes, permitting the acquisition of full 24 bit resolution data, a "fixed gain" 16 bit mode, or a "pseudo floating point" 16 bit mode.

Figure 5 is a block diagram of the AD24 board functions while Figure 6 is a more detailed diagram of the differential receiver for each of the 8 channels. The amplifier, an Analog Devices AD620, is an instrumentation design. We have it configured to receive either a voltage or current mode signal. It is possible to supply power to a sensor such as a hydrophone or simply connect an unpowered sensor such as a geophone directly. The current mode connection, in our case for a hydrophone, develops the differential signal across the 2 - 200 ohm resistors, AC coupled to the amplifier inputs. The additional 402 ohm resistor and the large 680  $\mu$ F electrolytic capacitor effectively "stiffen" the sensor supply voltage against low frequency modulation of the power supply by other sensors which can otherwise appear as crosstalk. This design emphasized rejection of crosstalk between signals in long unshielded twisted pair (UTP) cables. The crosstalk data cited in Figure 6 are an indication of what can be achieved using this current mode sensor/preamp arrangement.

Solder jumper pads allow a preamplifier gain of 0db, 20db, or 40db to be set, resulting in full scale input ranges of +/-4.5v, +/-0.45v, and +/-0.045v respectively. It is convenient to have a maximum sensor output voltage of about .9v peak-to-peak with a preamplifier gain of 20 dB which realizes most of the CMR available from the AD620 and also (for our hydrophones), causes the output referred noise of the analog front end to be comparable to the ADC "shorted input" noise level.

After an initial configuration sequence by the host PC processor, each AD24 board operates independently, with 16 KB of first-in/first-out (FIFO) data buffer space. One AD24 operates as the MASTER, jumpered to interrupt the host when data reaches the FIFO half-full mark. All are driven by a common clock and all respond in parallel to the same I/O command to synchronously start the sampling process after configuration procedures have completed. The clock must be clean and free of glitches, reflections, etc. A nasty clock line can result in what appears to be loss of data FIFO sync among multiple cards, bad status bits and an overabundance of data interrupts.

Each AD24 card in a system is programmed from the host PC. A FIFO on each board is used as a sequencer to control the mode and operation. At run time the processor downloads setup information and sequencer code to each board via I/O. A brief discussion is given here however a detailed explanation is provided in Appendix A. The programmable parameters are:

- SAMPLE FORMAT, one of 3 choices: 1) 24 bit mode, 2) 16 bit psuedo floating point mode, and 3) 16 bit fixed point mode.
- ACTIVE CHANNELS, selectable from 1 to 8. (Note: In the 24 bit mode the number of channels selected must be even - because of hardware restrictions.)
- SAMPLE RATE: There are 7 output rate selections given by "codes" 0 - 6, as a function of

the CLOCK rate which can be between .5 and 1.5 MHz. When discussing S-D converters, a distinction is usually made between sample rate and output rate because the actual sampling rate of the input signal is a multiple of the data output rate. All channels on an AD24 must sample at the same rate.

- SAMPLE COUNT INTERRUPT: Eight bits of a divide-by-n counter setting an interrupt rate that can be used to mark time at data record boundaries.

### 3.2.1 24 Bit Mode

After all 8 ADC's on a card complete a conversion, data from each ADC are shifted out serially, sign bit first followed by the most significant bit (MSB), into an 8 bit serial-to-parallel register. When the first 8 bits have been shifted into the register, they are stored in FIFO "A". The second 8 bits are shifted out and loaded into FIFO "B". The final 8 bits are shifted into the same register, and loaded into the next location of FIFO "A". Since there are not 3 data FIFOs, the 3 bytes of data are spread over 1-1/2 data locations, which is the source of the requirement for an even number of channels in 24 bit mode. Therefore 24 bit mode uses 50% greater storage and bus bandwidth compared to the two 2-byte sample modes.

### 3.2.2 Pseudo Floating Point Mode

Subsequent to conversion by all enabled ADC's on a board, the output word is formed by extracting a 14 bit signed mantissa from the 24 bit original sample based on the occurrence of the first "active" bit in the sample. The 2 remaining bits of the 16 bit output word become an exponent specifying up to 3 - 3 bit left shifts that determine the position of the mantissa in the original 24 bit format. In this way, the 16 bit output format can span the entire measurement range by simulating the output of an auto-ranging amplifier with 4 discrete gains. A detailed explanation is in Appendix A.

### 3.2.3 Fixed Point Mode

A second 16 bit mode was designed to select a fixed point 2 byte word from a fixed position in the 24 bit ADC output, including the sign bit. The rationale for this mode was the extra 2 bits of resolution compared to the floating point mode when a 16 bit output word is necessary. Unfortunately the design is flawed in that no provision was made to force the resultant 2's complement word into an over-ranged condition when any bit more significant than the output group is set. Thus this mode is limited to selection of the uppermost 16 bits of the original 24 bit value.

### 3.2.4 Data Storage

The AD24 board has a data buffer of 8 K 16-bit words using 2 8-Kbyte FIFO's, allowing a relaxed host response to data interrupts. The FIFO's are in sockets permitting more storage with larger devices of the same pinout. Using 8 channels, up to 512 sample suites can be offloaded by the PC at each interrupt from the FIFO half full flag. At a sample rate of 3906 Hz, this occurs every 131 ms for 16 bit data and about every 87 ms for 24 bit data.

Jumpers are available to allow selection of the appropriate interrupt request line (the selections are 3,4,5,10,11,12 & 15). A 33 MHz '486 typically initiates interrupt service in 15 - 20 microseconds and 16 bit data are transferred using the assembly instruction "INSW" with the REPEAT prefix at about 1.1 MHz. A half-FIFO of data can be offloaded in about 4 milliseconds. As will be discussed later, this offload period was a troublesome symptom with respect to noise in the data.

In 24 bit mode, a data suite consists of 8 3- byte words, (24 isn't integrally divisible into 4096), making it necessary to unload less than half a FIFO of data. We have used 4092 words which is 341 sample suites with 8 channels.

When data from the last channel is being stored for any particular sample suite, a marker bit is written to the ninth bit of FIFO "A" (these FIFO's are actually 9 bit devices), as a demarcation between samples suites. When data are read by the PC, this bit is written to the system status register and can be used to check that data frames have not been fractured.

### 3.2.5 Sequencer

A third FIFO serves as a sequencer to provide programmable signals to extract ADC data and temporarily store it in the data FIFO according to the selected sample format. The system clock is used to sequentially read out the pattern of bits downloaded by the host during configuration, resulting in 9 synchronous signals to control the timing of the sampling process. The actual sampling process of each ADC is continuous. With each new sample suite the sequencer cycles, completing a readout and load to FIFO of data from each active channel in  $24 + 25*n$  CLOCK cycles ( $224\mu s$  for 8 channels with a 1 MHz CLOCK in *pfp mode*). Figures 7 & 8 show timing details for the 24 bit and pseudo floating point modes. The "C" code used to create sequencer code for the 24 bit and pfp modes is given in the routine *ad24seq*, Appendix C.

### 3.2.6 Use of Multiple AD24's

Two AD24 cards were used in each RLAM. However more can be used to achieve systems with a large number of channels given adequate I/O space. A practical limit of about 10 boards in a single backplane system is imposed by ISA bus drive/loading capability. Unfortunately, we gave insufficient consideration to bus loading in our design as each AD24 has a number of devices attached directly to the ISA data and address bus lines rather than a single set of buffers. Despite this we have been able to run as many as 9 boards on the same backplane.

All cards have the same address to coincidently command the start of conversions. Theoretically, when using multiple AD24 boards, different sampling rates could be programmed if the acquisition software accommodated the appropriate amount of data from each board at the data interrupt rate.

### 3.2.7 Sample Count Interrupt

A sample count function exists on the AD24 whose output on rollover can provide a second interrupt to the host that can be used to mark the time of the first sample in the next record. The function is implemented with jumper selection of the upper 6 bits of a 12 bit counter and an 8 bit divide-by-n counter whose count is programmable by the host. The signal *DRDY1* is the input to this counter chain. The number of sample suites signaled by this interrupt is given by  $2^m * n$ , where  $m$  is the jumpered power-of-2 output and  $n$  is the 8 bit divisor programmed into the divide-by-n counter. For RLAM operations AD24's were jumpered for an  $m$  of 9, yielding a modulo of 512. When 3 RLAM's were in use,  $n$  was set to 21 to yield 10572 sample suites from 16 channels between interrupts. This event was used by the RLAM processor as an interrupt to read the realtime clock, indicating the time, typically to within a few tens of  $\mu s$  of the first sample in each data record.

### 3.2.8 Power

The AD24 board requires 5 power voltages, the sources for which can be configured in multiple

ways. The Crystal ADC chip set is comprised of 2 devices. The modulator chip, CS5321 has separate analog and digital power pins requiring 2 sets of +/- 5V. We use the same +/- 5V for both, separating them with a 10 ohm resistor and bypassing capacitors. (After the board was layed out, it came to our attention that the 2 pairs of power pins on the CS5321 were connected internally so the 10 ohm resistors do nothing.) The CS5322, the FIR decimator/filter, has pins brought out for two +5V supplies. The differential amplifier, sensor power and broadband buffer section of the board require bipolar power (+/-8 but can range from +/-5 to +/-12V). The remainder of the board consists of digital devices requiring +5V power with no particular noise restrictions. Separate power and ground planes for the analog, ADC and digital logic sections are provided, with connection choices as shown in Figures 9 and 10. This reduces signal contamination from power sources.

Through a series of jumpers the power connections can be configured in various ways depending on the available voltages and the requirement to minimize power dissipation. Normally, the +/- 5v for the ADC's come from a pair of linear regulators supplied with +/- 12v from the ISA bus. Similarly +/- 8v for the analog section comes from a second set of passive regulators. The +5v for the digital logic comes directly from the ISA bus. Alternatively, the analog section may be powered by an external +/- 8v source brought in through the signal connector. From this +/-8v, the +/- 5v for the A/D can be regulated. The +5v for digital logic can be supplied externally via the signal connector also. The intention of these external power sources is to allow some flexibility to minimize noise contamination.

Regarding noise rejection, the passive regulators are clearly beneficial when power is taken directly from the ISA bus +/-12V. They could be bypassed if external supplies are used though linear regulators on each board offer isolation and local filtering. Tests indicate that data noise levels are similar using power via the passive regulators and the ISA bus or from separate battery sources for the analog and ADC sections. We attribute this partly to separation and reasonable connection of ground planes in the 6 layer AD24 board.

Through various powering schemes, we have taken measurements of the actual current required by the major components of the AD24 board. These data (in millamps) are shown below for 8 channels @ 1 MHz CLOCK from board S/N 8:

	+8A	-8A	+5A	-5A	+5D
AD620's, OP220's, 4.5V Ref,no sensors	10.25	8.10			
CS5321/2 chip sets, includes LM317,337			48.20	48.40	
Logic +5, int clock @ 1MHz, no data					12.6
Logic +5, int clock @ 1MHz, taking data @3906Hz					35.0
Logic +5, ext clock @ 1MHz, no data					13.2
Logic +5, ext clock @ 1MHz, taking data @3906Hz					33.7
Logic +5, ext selected, no clock connected					1.0

Drawing power from the ISA +/-12 and +5 buses, each AD24 card dissipates about 1.5W clocked at 1 MHz with no sensors connected. If regulated power is supplied directly to the +/- 5A and +/-8A buses on the card, about .8W would be dissipated (at the AD24) under the same conditions. This is due to the losses of the LM3x7 linear regulators.

To prevent the input stages of the ADC's from latching up from excessive current flowing into signal inputs, the application of +/-8v to the analog section is delayed relative to the ADC's. Resistors R10 and R11 ( $604\Omega$ ) prevent excessive input current to the A/D should the AD620 become over-ranged on large signals. The combination of R10 and R11 with C6 and C7 also provides low pass filtering necessary to prevent aliasing at multiples of the S-D modulator clock.

Figure 10 shows the possible jumpering schemes for supplying power to the AD24 board. Any of 3 voltages +V, -V, & +5 can be brought in via the DB37 connector at the card edge. Likewise, the same voltages can be taken directly from the ISA bus where +V=12 and -V=-12.

### 3.2.9 AD24 Performance

A potential problem with a high resolution ADC in close proximity to a computer and bus is the corruption of the analog signals by radiation from high speed switching signals.

During tests, the AD24 board displayed a susceptibility to noise pickup that appeared to be related to I/O operations when data was read from the FIFO. Figure 11 is a sample of data taken early in the SIMI-94 experiment showing the symptom of this problem. These noise spikes can vary in amplitude from channel to channel and card to card. Tests indicated that the noise peaks occurred during the time that the FIFO's were being read and to a lesser extent, when I/O operations were being done with other devices on the bus, clearly evident in Figure 11. After testing, we concluded that the noise was not picked up on the board itself but rather by improperly shielded signal leads attached to the DB37 pin connector at the board edge. The symptom we observe is from induced high frequency interference (probably harmonics of address and data signal transitions on the ISA bus and the AD24 board data FIFO's), via signal leads, which is then nonlinearly rectified at input junctions of the AD620's. The amplifier has little common mode rejection of such high frequency signals. We found that foil shielding on the signal leads up to the solder pockets on the DB37 connector greatly inhibits the induced noise. We also found that the noise was eliminated if 8 or 10 turns of a shielded signal cable was put on a small toroid close to the point where the signal was connected to the AD24 differential amplifiers. Implementation of the latter measure was not practical, however another approach was nearly as effective. Surface mounted 470 pf NPO capacitors retrofitted across bias resistors R5 and R6 shunt the high frequency interference before they get to the AD620 inputs. In all cases this nearly eliminates any of the characteristic pulses in the signal. Figures 12 and 13 show data from the same channel before and after being fitted with the extra bypass capacitors.

Subsequent to noise reduction modifications, tests were run on AD24 boards to compare the fidelity of the 24 bit and "pfp" modes. Plots were made of data taken with signals input directly to the SDC and input normally to the differential receiver using a current mode hydrophone preamp/driver having the sensing element replaced with a capacitor. Tonal signals were input to the preamp using a battery powered KH4400A signal generator which is capable of S/(N+D) ratios of about 100+ dB. Not only is it difficult to perform such tests with little 60 Hz contamination but signal sources capable of meaningful tests of these converters are scarce.

Plots are included of tones input to an AD24 board in 2 ways. The first group, consisting of Figures 15a-h, 16a-h & 17a-h, shows the spectral result with a tone input directly to the 24 bit converter via a jumper pad on the board that bypasses the front end differential amplifier. This is not a normal input mode but is intended to give baseline data for the amplitude response of the SDC itself. Spectral levels have been normalized to dBV at the SDC input. We cannot say whether the distortion and noise levels are a function of the SDC or of the signal generator. Engineers at Crystal Semiconductor Corp. claim to have the same difficulty, although their plots look a few dB better than ours. Figures 15a,c,e,g show 24 bit data and Figures 15b,d,f,h show "pfp" mode 16 bit data. The same is true for Figures 16 and 17. Each plot is an average of 8 - 1024 sample frames and the spectral levels have been normalized to show the tonal peak correctly in dBV. A 7 point Hody window was used to minimize sidelobes.

The second group of plots, Figures 18 - 23 show data plots when a current mode hydrophone preamplifier drives the AD24 input with input shorted and with tones. The preamp gain was 28

dB, a typical gain with a ceramic cylinder sensor element. These plots are of single 1024 sample frames with no averaging. Spectral levels have been normalized to dBV and PSD at the board input.

The following plots are included:

1. Figures 15-a,c,e,g are 24 bit, output rate of 976 Hz, 400 Hz BW
2. Figures 15-b,d,f,h are 16 bit (pfp mode), output rate of 976 Hz, 400 Hz BW.
3. Figures 16-a,c,e,g are 24 bit, output rate of 1953 Hz, 800 Hz BW
4. Figures 16-b,d,f,h are 16 bit (pfp mode), output rate of 1953 Hz, 800 Hz BW.
5. Figures 17-a,c,e,g are 24 bit, output rate of 3906 Hz, 1600 Hz BW
6. Figures 17-b,d,f,h are 16 bit (pfp mode), output rate of 3906 Hz, 1600 Hz BW.
7. Figures 18a,b are 16 bit (pfp), 100 Hz tone, 400 Hz BW.
8. Figures 19a,b are 16 bit (pfp), 100 Hz tone, 800 Hz BW.
9. Figures 20a,b are 16 bit (pfp), 100 Hz tone, 1600 Hz BW.
10. Figures 21a,b are 16 bit (pfp), Noise floor PSD, AD24 i/p open & preamp i/p shorted, 400 Hz BW.
11. Figures 22a,b are 16 bit (pfp), Noise floor PSD, AD24 i/p open & preamp i/p shorted, 800 Hz BW.
12. Figures 23a,b are 16 bit (pfp), Noise floor PSD, AD24 i/p open & preamp i/p shorted, 1600 Hz BW.

Ignoring distortion and what appears to be sidelobe energy, plot 16a of the 24 bit data shows about 114 dB rms dynamic range (19 bits) over 400 Hz. The "pfp" mode result in Fig 15b shows about 80 dB rms dynamic range over the same bandwidth which is quite acceptable given the 13 bit mantissa in this mode. The distortion peaks in the pfp result (15b,d,f) are due to the truncated mantissa.

The noise level of the AD24 board with an open input (Figures 21a, 22a, & 23a) is 20 to 30 dB below what is shown with a +28dB hydrophone amplifier, input shorted (Figures 21b, 22b, & 23b). This assures that the AD24 board with +20 dB gain will not be the limiting factor in the system noise floor. In fact, the same can nearly be said of the AD24 with unity gain at the differential receiver (from Figures 15e,g). In order to keep the power requirement low (about 45mw), our current mode hydrophone sensors have a maximum linear output of about -10dBVRms. Since the input range of the converter is +/- 4.5V, (10.05 dB), setting the gain of the AD620 at +20dB is a good match for maximum signal levels and uses most of the available CMR of the AD620 as well.

The hydrophone preamp is the source of the distortion levels evident in Figures 18 - 20. It is difficult to achieve better than about -60dB signal-to-distortion ratios with a low power current mode driver such as we have used.

### 3.3 Telemetry

Two radio systems were used. The data telemetry link was a commercial "wireless" Ethernet LAN adapter for the ISA bus (WaveLAN) from NCR (now from AT&T) operating in the 902-928 MHz Industrial Scientific and Medical (ISM) band. It is a spread spectrum link specified to have a maximum throughput of 2 Mbits/sec. Experiments over short ranges (few hundred yards) have resulted in maximum rates of about 180 KBytes/s. The output radio power of the WaveLAN board is specified at 250mw (24 dBm) which when coupled with a looped YAGI directional antenna with about 12(db gain) about 16' off the ice, was more than adequate to reach the 4 km distance to camp. The antenna at the camp was a 9 dB omnidirectional "stick" about 10 feet long, (Decibel M/N DB589) mounted atop a 30 foot triangular cross-section tower. A 20 foot length of RG8 (about 1.5 dB loss) was used to connect the WaveLAN to its antenna at the RLAM's. At the camp, a Hyperlink 900-U power amplifier/preamplifier was mounted at the base of the antenna. This unit combines a low noise preamplifier on input (+22 dB gain) with a power amplifier on output (+24 dB) and an automatic T/R switch which senses when RF power is sourced, switching transparently in < 3 $\mu$ s. With about 60' of RG58 between the WaveLAN board and the antenna, not only was the net RF power applied to the antenna boosted to about 2.5 W but cable attenuation was eliminated by the preamplifier gain. Power to this unit is supplied from the coax cable via a "T" coupler with an external power supply at the inboard end.

The RF requirements of this system are straight forward; minimize losses in cables and connectors (though little is lost in a good connection), maximize antenna gain and height.

A second "survival" radio link (Figure 14), was implemented with a UHF (455 MHz) FM radio (a Motorola Handy-Talkie) with the standard FM narrow bandwidth of 3 kHz. This was operated at each RLAM in a receive only mode, allowing simple DTMF codes from a transmitter at the main camp to power and unpower RLAM's remotely. All RLAM's were configured to respond to the same code to turn off/on the DC/DC converter that supplied +5V and +/-12V power from a single battery pack. The UHF radio and DTMF decoder ran off passively regulated power directly from the battery pack.

In addition to providing a failsafe reset method for the RLAM computer, the UHF radio was used to supply the RLAM's with IRIG-B timecode. Consequently, all RLAM's were synchronized to a common absolute time source (GPS time) and run on a common time base derived from the 1 kHz IRIG-B carrier. The timebase for each RLAM was a 1 MHz voltage controlled crystal oscillator (VCXO, Oscillatek, M/N HCM8304D50B, about \$100) whose control input for a +/- 50 ppm servo range was the filtered output (.6 Hz single pole RC LPF) of a phase comparator. The phase comparator inputs were the 1 KHz timecode carrier and the VCXO output divided by 1000. Hence the short term stability of the timebase was a function of the local VCXO (and filter) but the longterm stability was controlled by the 1 MHZ of the GPS receiver at the camp. Conveniently, the same GPS receiver timebase was used by acquisition systems at the camp for the horizontal and vertical arrays. Figure 14 is a schematic showing the timing control circuitry.

During much of the last week of the experiment, 3 RLAM's were run at a sample rate of either 976 Hz or 1953 Hz. The higher rate (187,392 byte/s) could not be maintained, however, with 14 MB of data buffer space in each RLAM, we could get continuous streams of data from 10 to 15 minutes in length before the system would suffer from data overflow. At this higher rate, the receiver was programmed to restart all 3 RLAMs's when any one of them overflowed.

There is an as yet unresolved problem with the LAN telemetry process which is probably software. Regardless of the number of RLAM's in use, there were times when the receiving machine would stop logging data with 1 of 2 errors reported:

- "net read error: NET ERR NOTESTAB" , which implies that the RLAM that was next in

line to send data acted as if it's STREAM connection had been dropped.

- "Bogus REC HDR KEY" which implied that the 32 byte header that had been received was not in fact a header, i.e. the RLAM either didn't send it or somehow the receiving machine got confused about what it should be receiving.

Changes to RF components of the system had no effect. There were occasions when 1 or more RLAMS ran continuously for more than a day, as well as other times when 1 might run for only a few minutes. Clearly, attention to this problem is needed prior to future use of this system.

### 3.4 Receiving & Storage System

The data receiving system is shown in Figure 24. It consisted of the 9 dB omnidirectional antenna, HyperAmp, RG58 connecting cable, a PC equipped with a single WaveLAN Ethernet adapter and an 8mm tape drive.

Data were stored in a sequential series of files on 8mm tape in a multiplexed format with each file consisting of an arbitrary number of records. The data block size was always 1024 bytes and standard filemarks were used at the end of files, including the 1024 byte tape header. The structure of a tape header is:

```
struct tape_header
{
    unsigned char dhkey[4]; /* tape header key, "MTHD" */
    unsigned char sn[16]; /* Tape S/N, 15 chars terminated */
                           /* with 00h */
    unsigned int dhtime[4]; /* date/time header was written */
                           /* * year, unsigned int; month/day, 2 packed unsigned chars */
                           /* * hour/minute, unsigned int; second/millisecond, unsigned int */
    long chblk; /* 0 */
    long dhblk; /* 0 */
    char ayear[16]; /* ascii 1994 */
    char amonth[16]; /* ascii */
    char aday[16]; /* ascii */
    char ahour[16]; /* ascii */
    char aminute[16]; /* ascii */
    char blank[900]; /* 0 */
    long dhcont; /* 0 */
    unsigned char dhkey1[4]; /* tape header key, 'MTHD' */
};
```

A SIMI-94 RLAM data file is a contiguous sequence of records typically 1009 Kbytes in length. Each begins with a 1024 byte Data Record Header (DRH), in which the record size, number of RLAM's, channels, etc are identified. Most DRH information is constant throughout a file however time, record number, buffer number and other status for individual RLAM's is updated regularly. An example is shown as Table 2. With a single RLAM in use, each record consisted of the same number of 16 ch sample suites, seamlessly connected to the data section of the next record. If more than 1 RLAM was in use, the data from one RLAM was followed by data from the 2nd RLAM (and 3rd if in operation) within each record. The data format on tape is:

1024 byte tape header

```
graph TD; Root[Data file 0] --> Record0[Data record 0]; Root --> RLAM0[Data from RLAM 0]; Record0 --> Suite0["Suite 0 thru suite \"m\""]; RLAM0 --> RLAMn1[Data from RLAM n-1]; RLAM0 --> Suite1["Suite 0 thru suite \"m\""]; RLAMn1 --- RLAMn1_VLine[ ]; Record1[Data record 1] --> LastRecord[Last data record]; Record1 --> EOF1[EOF]; LastRecord --- LastRecord_VLine[ ]; EOF1 --- EOF1_VLine[ ]; subgraph DataFile1 [Data file 1]; EOF2[EOF]; LastFile[Last data file]; EOF2 --- EOF2_VLine[ ]; LastFile --- LastFile_VLine[ ]; end
```

The DRH format is a structure as shown below. It gives the time to the microsecond of the first sample in the data block that follows and the record number. Subheaders from each of the RLAM's are embedded in the "other[]" array which give information that is diagnostic in nature and offers a measure of confidence that the data acquisition process has proceeded correctly.

```

struct data_rec_hdr
{
    unsigned char    rhkey[4];      /* header key, "DATA"      */
    unsigned char    proj[16];     /* project name, ascii    */
    unsigned char    extype[32];   /* exp type, ascii        */
    unsigned int     exp;          /* exp number             */
    unsigned int     date[2];      /* RLAM date...(year, Jday) */
    unsigned int     time[2];      /* RLAM time...(minutes, ms) */
    unsigned int     ch;           /* # channels             */
    unsigned int     bkch;         /* # blocks per demuxed channel */
                                /* if zero, data is not demuxed */
    long             npts;         /* # sample periods        */

```

```

float      rhfs;          /* sample rate in Hz          */
unsigned int rlen;          /* record length in blks, includes    */
                           /* /* sector used for record header */
                           /* */
unsigned int rec;          /* number of record that follows */
                           /* */
char       rhlat[16];      /* lat, ascii DDD MM SS.T N or E */
                           /* */
char       rhlng[16];      /* long, ascii DDD MM SS.T E or W */
                           /* */
unsigned char lpf[128];    /* unused */
                           /* */
unsigned char pag[128];   /* unused */
                           /* */
unsigned char hpf[128];   /* unused */
                           /* */

unsigned char mrh0[32];    /* 32 bytes of RLAM 0 header */
                           /* */
unsigned char mrh1[32];    /* 32 bytes of RLAM 1 header */
                           /* */
unsigned char mrh2[32];    /* 32 bytes of RLAM 2 header */
                           /* */
unsigned char mrh3[32];    /* 32 bytes of RLAM 3 header */
                           /* */
/* 32 byte RLAM header(s) copied to mrh0 - mrh3 containing: */
/* char      key[4];        'LTX#' where '#' = RLAM ID */
/* unsigned int rec;        record number */
/* unsigned int yr;         1994 */
/* unsigned int dy;         day */
/* unsigned int mh;         hours*60 + minutes */
/* unsigned int ms;         seconds*1000 + milliseconds */
/* unsigned int mc;         microseconds */
/* unsigned int batt;       battery voltage, unused */
/* unsigned int temp;       internal temperature, unused */
/* unsigned char op_flag;   output buffer number */
/* unsigned char ip_flag;   input buffer number */
/* unsigned char ovf;        overflow counter r */
/* unsigned char scan_blocks; number fifo buffers/rec */
/* unsigned char rate_code; AD@4 digitization rate code */
/* unsigned char nch;        number of channels used */
/* long       xmsrc;        absolute adrs of o/p buffer */
/* unsigned int data_error; error flag, indicates XM prob */

int       slew_delay;      /* unused */
                           /* */
unsigned int microsec;    /* 3 dec digits in microsec, < 1000 */
                           /* */
unsigned int sample_mode; /* 0 = normal, 1 = offset updated */
                           /* */
unsigned int preamp_gain; /* linear fixed gain than ranged */
                           /* */
unsigned int sample_period; /* sample period in microsec */
                           /* */
unsigned int buffer_num;  /* buffer number stored this record */
                           /* */
long      rectime;        /* record time in microsec */
                           /* */
unsigned char other[366]; /* space left in record header */
                           /* possibly used for sensor locs */
                           /* */
char      fname[16];      /* Data file name, new for Zircon */
                           /* */
unsigned char rhkey1[4];  /* end of rec header key "DATA" */
                           /* */
};


```

As described earlier, the *pseudo floating point* mode was used, resulting in 2-byte samples stored as *unsigned short intergers* with the most significant byte appearing first in a sequence of bytes

taken from a tape. The bits are positive (high logic level) true with the bit assignments as shown:

BIT	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
	SN	M12	M11	M10	M09	M08	M07	M06	M05	M04	M03	M02	M01	M00	G1	G0
{+/-}{	13 BIT MANTISSA													}{["GAIN"]}		

RLAM data was acquired in 4 different configurations.

RLAM's	SAMPLE RATE	REC LEN ON TAPE	PTS/REC	RLAM TX SIZE	NET DATA RATE
1	3906 Hz	1009 KB	32256	1032192 bytes	125,000 bytes/s
2	1953	993	15872	507904	125,000
3	1953	1009	10752	344064	187,500
3	976	1009	10752	344064	93,750

Approximately 120 GB of data was collected using these combinations of 1 or more RLAM's and stored on 8mm tape using an Exabyte 8500 or 8505 tape drive. We used a Rancho RT10-AT SCSI host adapter. The feature that originally made this adapter useful to us is its use of DMA for data transfers between the ISA bus and the adapter. The adapter BIOS includes a direct-to-SCSI function that accepts a 32 bit absolute address that it converts to a 16 bit DMA address and 8 bit page address, allowing it to directly access the lower 16 MB of memory space. The adapter can transfer large buffers (typically about 1 MB) with a single command, minimizing host interaction during transfers to tape.

### 3.5 Power

The power requirement of the RLAM system as configured for SIMI-94 is summarized as follows:

DEVICE	POWER DISSIPATION (watts, measured)
Teknor AT-4 SBC	7.5
WaveLAN radio LAN board	8.5 idle, 11 transmitting, 9.5 avg
Time base/survival radio/DTMF receiver	1.0
2 AD24 boards @ 1 Mhz, digitizing	3.0
DC/DC converter overhead	4.0
sensors (geophones)...	negligible
 TOTAL power draw from battery while acquiring and telemetering data:	 25 W

While lithium batteries are superior to both lead-acid and alkaline with respect to low temperature and kwh per volume and weight, they weren't cost effective for this application where power density was not the primary consideration. They are still about 5 times as expensive as alkaline per kwh.

For past Arctic experiments, we have occasionally used automotive batteries filled with a richer acid solution in temperatures as low as  $-30^{\circ}\text{C}$  inside a "cooler" but drawing only about 5 watts. For the RLAM's we considered lead acid gel batteries designed for low temperature operation however the power/weight ratio is only about half that of alkaline at  $0^{\circ}\text{C}$ . Recent data on Duracell alkalines shows that nearly 75% of a cell's rated power at  $70^{\circ}\text{C}$  is available at  $0^{\circ}\text{C}$  (See Appendix B). We chose alkaline battery packs for the following reasons:

- We combined our purchase with another project to obtain pricing that made alkaline cheaper than rechargeable lead-acid gel batteries.
- the power density of alkaline is higher, even at  $0^{\circ}C$ , requiring fewer battery exchanges.
- Alkaline packs can be made to any configuration and voltage.
- no "hazmat" shipping concerns with alkaline batteries
- no concern with acid leakage inside RLAM case
- no battery charging requirement

Fourteen battery packs were purchased for \$160 each from Battery Assemblers Inc, (Bohemia, NY, 516 567 8855) each consisting of 144 alkaline "D" cells, with welded connections in a series/parallel, diode protected configuration as shown in Figure 25. Each pack was estimated to have about 1.2 kwh at  $0^{\circ}C$  starting at 36V. In retrospect, with the higher than expected battery temperatures, it may have been cheaper to use automotive style "sealed, low maintenance" batteries.

For some of the RLAM deployments, the temperature at the top of the battery packs and/or at the side of the chassis in contact with the CPU heatsink was recorded at a regular rate using Onset HOBO temperature loggers. Figures 26a and 27a show the temperature during one deployment, at the processor and at the top of the battery over the same time period (despite different sample periods of 4.8 and 6.4 minutes). Figures 26b and 27b are spectra of the same data. As with all the data, there was a noticeable bump in the spectrum at the "daily" frequency, suggesting that the internal temperature tracked the sun. We were at  $74^{\circ}$  latitude in April, so the sun was below the horizon for 8 to 4 hours per day over the course of the experiment. The lower frequency trend in the temperature appears to have tracked the ambient air temperature shown for the same time in Figure 28a. Its spectrum is shown in Figure 28b, confirming a peak at the daily interval. In both time series plots the point where the unit was brought into the hut (still running) is shown.

Our interest was peaked by the high processor heatsink temperatures. Considering the insulation and the inner dimensions of the RLAM case, we had estimated conductive thermal losses of  $.92 * \Delta T^{\circ}F$ , which for a  $90^{\circ}F$  differential (inside @ $60^{\circ}F$ , outside at  $-30^{\circ}F$ ), suggests a loss rate of about 83 BTU/hr. The 25 W dissipation of the electronics was equivalent to 88 BTU/hr. This near equivalence is probably fortuitous since the internal space was very much decreased by the snug fit of the batteries and convective losses through the hole in the case, despite a nylon cover, were surely significant.

A single DC/DC converter (Intronics KZ338, 40W max, 20-60V input) provided power at +/- 12V and +5V to RLAM electronics. Our conservative expectation was that 2 alkaline battery packs would provide at least 4 days operation. In fact, the insulated cases kept them warm enough that they remained operational for 8 days. Even then, in one case, the pair of battery packs was down to 23.2V under load, which meant that there was still about a day left. In practice at the higher than expected temperatures, it seems that each pack can provide as much as 2.5 kwh.

### 3.6 Software

The software consisted of 2 main programs, TXRLAM which ran on the RLAM's and RXRLAM which ran on the receiving computer. Both machines ran DOS 5.0.

In the field the RLAM's were configured to boot from 1 MB of flash memory as drive A:. For software development purposes, they could be connected to hard and floppy disks via the onboard SBC interfaces and booted normally. A switch on the side of the chassis controlled the effective

boot drive. No memory manager was used since all of extended memory was accessed via interrupt 15 services for data buffering. Unessential utilities, TSR's or drivers were not used, to keep the system as simple and unencumbered as possible.

Power was applied to the RLAM's, either by mating the 2-pin connectors that were available at the 4" diameter cable entrance or by the survival radio link switch which controlled the shutdown input of the DC/DC converter. The "autoexec.bat" file would continuously invoke the application "ftp" over the WaveLAN link in an attempt to download a program of a particular name from the receiving system. If the receiver was alive, this program would be downloaded and executed. Normally it was TXRLAM but could have been any other program as well. Once the program was aboard, the RLAM would execute it. This feature allowed TXRLAM code to be modified at any time to fix bugs or add features and simply be made available the next time the RLAM's were forced to power up and boot via the survival radio link. When not telemetering data, the RLAM's would repeatedly attempt to establish a STREAM (TCP/IP) connection over the WaveLAN with the receiving machine. Once the connection was established, the RLAM would transmit its time at second marks to the receiving system and listen for a command block from the receiving computer.

The sequence of events to establish a TCP connection over the wireless LAN after power up was the following:

- RLAM's use "ftp" to attempt program download until successful
- RLAM's make repeated attempts to net\_connect with the receive system.
- At the receiver node, for each RLAM, a net\_descriptor is created, net\_listen is attempted for a net\_connect by each RLAM, desired net\_options are set.
- sequentially a net\_write requested of each RLAM for its time, net\_read, and shown on screen as a sign that the RLAM's are alive and well.
- When desired, a parameter command string net\_write to each RLAM, i.e. setup its acquisition program and commence data collection at a specified time.
- Sequentially net\_read data blocks from RLAMs. As data arrives, assemble a Data Record Header and write a data record to tape. The intent is to minimize transmission collision overhead by ensuring that RLAM's don't talk until spoken to.
- When operations are to be stopped, sequentially close connections. The RLAM's resort to net\_connect attempts awaiting a restart.

The 32 byte command block consisted of:

BYTE	VALUE	BYTE	VALUE
0	"escape"	16	CH 0 select
1	ascii 'c'	17	CH 1 select
2	sync_second	18	CH 2 select
3	adc_rate_code	19	CH 3 select
4	adc_mode	20	CH 4 select
5	scan_blocks	21	CH 5 select
6	# channels	22	CH 6 select
7	ipts (hi)	23	CH 7 select
8	ipts (low)	24	CH 8 select
9	spread	25	CH 9 select
10	0	26	CH 10 select

11	0	27	CH 11 select
12	0	28	CH 12 select
13	0	29	CH 13 select
14	0	30	CH 14 select
15	0	31	CH 15 select

Once the 32 byte command block was received, the AD24 boards were configured and synchronized at the time specified by "sync\_second". In this way, multiple RLAM's started sampling at the same time.

If any one RLAM malfunctioned, the receiver would recognize an error condition and the acquisition process would halt.

Each RLAM was equipped with 16 MB of memory, of which 14 MB could be used as a large circular data buffer. The WRITE pointer was started at the 1MB boundary. Data interrupts from the MASTER AD24 board would cause a routine to unload a fixed amount of data (typically half or 4096 samples) from the FIFO on each AD24 board. This would take about 3.6 ms per 8 channel AD24 using inline assembly and the INSW instruction per card (16 bit transfers).

The programmable sample counter was setup to interrupt RLAMS as discussed earlier. The interrupt from this event would cause a "subheader", shown below, to be assembled and the transmission of this 32 byte structure as well as a data record to the receiving computer. Likewise, the program RXRLAM at the receiver would know by keeping track of the amount of data telemetered from each RLAM, when to expect these subheaders. This was essentially a consistency check to assure that RLAM(s) appeared to be operating correctly. Once all subheaders were received, a new data record was assumed and the DRH was written to tape, followed by a data record (usually 1008 Kbytes), as it was received from RLAM(s). Thus there was no need to buffer full data records at the receiving computer.

The 32 byte RLAM "subheader" structure is:

```
struct rlamrec_hdr
{
    char key[4];
    unsigned int rec;
    unsigned int yr;
    unsigned int dy;
    unsigned int mh;
    unsigned int ms;
    unsigned int mc;
    unsigned int batt;
    unsigned int temp;
    unsigned char op_flag;
    unsigned char ip_flag;
    unsigned char ovf;
    unsigned char scan_blocks;
    unsigned char rate_code;
    unsigned char nch;
    long xmsrc;
    unsigned int data_error;
};
```

From the receiving machine, the acquisition process could be stopped at any time, causing a graceful disconnection from the RLAM(s).

Software was written using Borland V3.1 C in the DOS environment. The PC/TCP Development Kit libraries from FTP (the company) now support both Microsoft and Borland compilers. Our code used the FTP libraries, "pclib" and "netlib".

A Network Data Interface Standard (NDIS) driver is supplied with the WaveLAN adapter making it easy to implement TCP/IP connectivity using Microsoft's Lan Manager. This is really a key feature of using a radio LAN adapter as a telemetry link between 2 data acquisition systems because it allows one to implement a simplified IP functionality directly in acquisition code. The established reliability of the LAN protocols mean that the hardware or software engineer is not concerned with error detection, correction and data retransmission issues. (There may be a "Clarkson" packet driver available for the WaveLAN adapter by now but we have not pursued the matter.)

An improvement that we plan for a new project is realtime retransmission of data to another workstation connected to the receiver over a cabled Ethernet.

#### **4 Acknowledgments**

The Sea Ice Mechanics Initiative, of which the work described in this report was part, was funded by a grant from the Office of Naval Research, Code 322HL, #N00014-91-J-1296.

## 5 List of Figures and Tables

Figure 1. Mock aerial view of SIMI-94 camp, color  
Figure 2. Block diagram of RLAM functional components  
Figure 3. Diagram of RLAM box and components  
Figure 4. Photograph of RLAM deployed, color  
Figure 5. Block diagram of AD24 functions  
Figure 6. Diagram of differential receiver showing jumper connections  
Figure 7. Sequencer timing details for 24 bit mode  
Figure 8. Sequencer timing details for 16 bit pfp mode  
Figure 9. Ground jumpers  
Figure 10. Power source jumpers  
Figure 11. MTV data sample from SIMI showing I/O noise  
Figure 12. single channel time series before bypass caps  
Figure 13. single channel time series after bypass caps  
Figure 14. RTC, timebase CLK, "SURVIVAL" radio receiver  
Figures 15. - 23. plots of tests tone and noise floor spectra  
Figure 24. Data receiving system block diagram  
Figure 25. RLAM battery diagram  
Figure 26a. Temperature inside RLAM at processor Figure 26b. Spectra of processor temperature  
Figure 27a. Temperature inside RLAM at top of battery Figure 27b. Spectra of RLAM battery temperature  
Figure 28a. Time series of ambient air temperature during RLAM temp measurements Figure 28b. Spectra of outside ambient air temperature  
Figure 29. AD24 Schematic  
Figure 30. 16 Bit Pseudo Floating Point word Mapping to 24 bit word  
Figure 31. 24 Bit ADC Word to 16 Bit PFP conversion  
Figure 32a. Specifications of CS5322 FIR Filter characteristics  
Figure 32b. Specifications of CS5322 FIR Filter characteristics  
Figure 33. Duracell "D" Alkaline cell discharge curves  
Figure 34. RLAM battery pack test plots, 0°C  
Figure 35. RLAM battery pack test plots, 70°C  
Table 1. AD24 board jumper list  
Table 2. Data Record Header sample

## References

[1] von der Heydt, K., Kemp, J., Lynch, J., Miller, J., Chiu, C.S., Array Data Acquisition with Wireless LAN Telemetry as applied to Shallow Water Tomography in the Barents Sea", Woods Hole Oceanographic Institution Technical Report #WHOI-92-44, December 1992.

## A Appendix, AD24 Functional Details

### A.1 Operation

Here the operation of the AD24 board is described in detail. In retrospect, control functions of the AD24 may have been accomplished in less space and with more flexibility using a programmable logic device such as an Altera or Xilinx PLD. At the time we thought it more expedient to use conventional "glue" logic rather than invest the resources needed to become proficient at PLD design. Having done this, I think that we could argue in favor of the PLD based design.

The table below summarizes the I/O addresses associated with the AD24 operation. Some addresses are used for actual data, control or status I/O while others are used as strobes. All are or are treated as 16 bit operations, though only the data FIFO reads and the writes to the sequencer FIFO actually involve 16 bit transfers.

#### INPUT / OUTPUT PORT ADDRESSING FOR THE AD24 BOARD

ADDRESS - IN HEX	R/W	OUTPUT PIN	FUNCTION
3 2 1 0			
0 0 0 0	0	W	U22-Y0 8 bit, load control register (CTRL)
1 0 0 0	8	W	U22-Y1 8 bit, load sample counter (COUNT)
0 0 1 0	2	W	U22-Y2 9 bit, load sequencer (SEQ)
1 0 1 0	A	W	U22-Y3 strobe, latch cnt interval (COUNT_SET)
0 1 0 0	4	W	U22-Y3
1 1 0 0	C	W	U22-Y5 strobe, main reset (BOARD_RST)
0 1 1 0	6	W	U22-Y6 strobe, reset sync pulse FF (SYNC_RST)
1 1 1 0	E	W	U22-Y7 strobe, clock data from sequencer manually (NPSTRB)
0 0 0 0	0	R	U20-Y0 16 bit, read data FIFO (DATA)
1 0 0 0	8	R	U20-Y1 8 bit, read STATUS register (STATUS)
0 0 1 0	2	R	U20-Y2 strobe, reset data FIFO's (DATA_RST)
1 0 1 0	A	R	U20-Y3 strobe, reset sequencer FIFO (SEQ_RST)
0 1 0 0	4	R	U20-Y4 strobe, reset ADC chipsets (ADC_RST)
1 1 0 0	C	R	U20-Y5 strobe, cycle sequencer once (NADVSQ)
0 1 1 0	6	R	U20-Y6
1 1 1 0	E	R	U20-Y7

A WRITE TO ADRS 300H INITIATES SAMPLING SYNCHRONOUSLY ON ALL CARDS (AD24SYNC)

Prior to use, solder jumper selections associated with ground, gain and signal routing (at the differential amplifier) must be made on the AD24 board. Power jumper selections, interrupt selections and the sample counter modulo must be made using shorting jacks. Table 1 is a list of all the solder jumpers and "wrap or jack" jumpers on the AD24 board along with their function.

The I/O BASE address is selected with a dip switch. One of 4 selections (000,100,200 or 300H) for the ADC SYNC address are made with bits S8 & S9 of the same switch. The DIP switch bit functions shown represent a base I/O address of 210H and an ADC SYNC address of 300H. Bit S7 must be in the "ON", (logic 0) position.

	S1	S2	S3	S4	S5	S6	S7	S8	S9
LOGIC "0" ON	*	*	*	*	*	*			
LOGIC "1" OFF	*			*		*	*	*	
FUNCTION....	A9	A8	A7	A6	A5	A4	A3	SA9	SA8

The board initialization sequence for RLAM use was the following:

- issue SYNC\_RST (npsyrs)
- issue BOARD\_RST (nprst)
- issue DATA\_RST (ndrst)
- issue SEQ\_RST (nsrs)
- load the sequencer using code as shown in AD24seq
- issue CTRL word (nwrctl) set for host control, CTL=1
- issue ADC\_RST
- configure ADC's using code as shown in AD24conf
- issue CTRL word, R/W=1, RSEL=1, others=0
- issue NADVSQ to run sequencer 1 complete cycle
- load sample count interrupt register; first load the count value into the 8 bit register, then latch it into the counter, then load the (count value -1) into the register to be used subsequent to the first record.
- do any other acquisition setup required such as interrupt enables
- issue DATA\_RST to reset data FIFO's, all cards
- issue AD24SYNC, start ADC's synchronously
- issue SYNC\_RST to all cards
- respond to FIFO half-full interrupts to acquire data

## A.2 Sequencer Functions

Figure 29 is a circuit schematic for the entire AD24 board. Device U26 is an 8 Kbyte, 9 bit wide FIFO, used as a sequencer to implement state machine control of the digitization process after being loaded with a series of words by the host as part of the AD24 configuration procedure. Sequencer "code" can be constructed to implement any of the 3 - AD24 operating modes with any number of channels. We have constrained the number of channels in the 24 bit mode to be even to achieve efficient FIFO storage with 3-byte samples. The sequencer code consists of a particular pattern of 1's and 0's as a function of mode and the number of active channels. At the "data ready" pulse edge (DRDY1), logic to sequentially read 9 bit words from the sequencer FIFO is enabled.

The final function of the sequencer code is to reset itself, triggered by the appearance of the FIFO empty flag. The readout logic then becomes dormant until the next positive edge of a DRDY1 pulse.

From the time of AD24SYNC, the ADC's run continuously. The sequencer becomes active only after DRDY1 triggers a data readout cycle. The output sampling rate is a function of the CLOCK (1 MHz for RLAM's) and the value of the rate bits in the CS5322 control register, as shown above. Although the Crystal ADC chip set has a "power down" mode, this feature has not been used, therefore unused channels do conversions but data is not read from them.

As shown in Figures 7 & 8, data readout from all 8 channels takes 224 $\mu$ s. At the fastest output rate, only a short time remains between the end of 1 readout cycle and the beginning of the next. A sequencer readout rate at the ADC CLOCK frequency requires the sequencer clock to be twice that rate. After the rising edge of DRDY1, FF U7B is set applying the CLOCK to U14A and U13, resulting in STROBE and NSTRBE that are double the frequency of CLOCK, and used to readout sequencer words at the CLOCK rate. Data words are sequentially read from each of the active ADC's. SCLK is taken from the sequencer via U24 and applied to the ADC channel selected with sequencer bits 5,6 & 7. Data bits are serially clocked out sign bit first followed by the most significant data bit, at the CLOCK rate into shift register U32. Flipflop U31A is used to latch the sign bit.

Configured for 24 bit samples (mode 1), as each group of 8 bits is loaded into U32, they are loaded into the data FIFO's in position such that each 3-byte data word occupies 1-1/2 FIFO locations as shown below for an 8 channel suite.

FIFO WORD#	FIFO high byte	FIFO low byte
0	CH 0 high byte	CH 0 mid byte
1	CH 0 low byte	CH 1 high byte
2	CH 1 mid byte	CH 1 low byte
3	CH 2 high byte	CH 2 mid byte
4	CH 2 low byte	CH 3 high byte
5	CH 3 mid byte	CH 3 low byte
6	CH 4 high byte	CH 4 mid byte
7	CH 4 low byte	CH 5 high byte
8	CH 5 mid byte	CH 5 low byte
9	CH 6 high byte	CH 6 mid byte
10	CH 6 low byte	CH 7 high byte
11	CH 7 mid byte	CH 7 low byte

Mode 2 involves a remapping of the 24 bit word into a 16 bit word consisting of a 2 bit exponent and a 14 bit mantissa, Figures 30 & 31. This is analogous to having an auto-ranging amplifier, capable of 4 gain levels corresponding to 0, 3, 6 & 9 bits of gain. The sequencer code implements this as follows for each ADC in use.

The first 8 bits are clocked out and as in mode 1, the sign bit is latched in FF U31A. As shown in Figure 30, 1 of 4 possible data mappings is possible. The sign bit is always installed in the most significant position of the upper byte of the resultant 2-byte word. If any of the upper 3 bits (not including the sign bit) are active, the exponent (BA) bits are set to "00" and the 8 bits in the shift register are written to the upper data FIFO byte (U18). After 8 more bits are clocked into U32, the exponent is put into the lower 2 bit positions and the resultant is written to the lower byte of the data FIFO (U19). If none of the upper 3 original data bits are active, after 3 more are clocked

out the same test is repeated. This process can occur up to 3 times, each causing the exponent to be incremented. If no active bits are detected in the upper 9, the exponent defaults to "11", the mantissa becomes the sign bit and bits 13 through 1 of the 24 bit sample word (bit 0 is never used). The resultant high and low bytes stored in the data FIFO.

Obviously, small signals in the presence of large signals will suffer using this *psuedo floating point* scheme since the small signal energy will be lost in the process of truncating to 13 bits of resolution. However the tradeoff is often acceptable since the storage requirements become more manageable and the measurement range remains the full scale of the approximately 120 dB RMS dynamic range of the converter.

The tables below describe the functions of the CS5322 FIR filter control port bits and the relationship between CLOCK, output sample rate and bandwidth.

BIT	FUNCTION	DESCRIPTION (bits entered serially, MSB-7 first)
7	PWDN	0 = run, 1 = power down mode
6	ORCAL	0 = inactive, 1 = do calibration
5	USEOR	0 = inactive, 1 = use calibration offset register
4	CSEL	0 = use modulator output, 1 = use external "TDATA"
3	RESERVED	
2	DECC	msb of rate select code
1	DECB	
0	DECA	lsb of rate select code as per table below

RATE CODE	SAMPLE RATE	O/P RATE	O/P RATIO	BW@ -3dB	BW@ ripple	DYNAMIC RANGE
110=6	CLK/4	3906	CLK/256	1587	1465 @.20	103
101=5	CLK/4	1953	CLK/512	805	732 @.04	118
100=4	CLK/4	976	CLK/1024	401	366 @.08	121
011=3	CLK/4	488	CLK/2048	201	183 @.10	124
010=2	CLK/4	244	CLK/4096	100	92 @.10	127
001=1	CLK/4	122	CLK/8192	50	46 @.10	129
000=0	CLK/4	61	CLK/16384	25	23 @.10	130

The plots of Figures 32a,b are taken from the CS5322 data sheet. They show the FIR filter amplitude and phase response and show that the group delay is a constant 28 output words.

## B Appendix, Battery Test Data

At low currents, we think of alkaline "D" cells as about 10 Ah (10-12 Wh) at  $0^{\circ}C$ . As of April 1994, the Duracell model MN1300 was upgraded from a rating of 14.25 Ah to 17 Ah with a constant load of  $4.7\Omega$  to .8V at  $70^{\circ}C$ . The RLAM packs did not include the new, higher rated cell. The RLAM's draw about 24 W, or at 30 V, about .8 amp. The RLAM batteries consisted of 6 strings of 24 cells. With 2 packs, at 30 V, a current of .067 amp was drawn from each of 12 strings (about  $18\Omega$ ), considerably less than the manufacturer's test current with a  $4.7\Omega$  load.

The amp-hour rating specified by Duracell is the result of a test whereby a  $4.7\Omega$  load is applied to a fresh cell at  $70^{\circ}C$ . The time it takes for the cell voltage to decrease to .8V is measured. During this test, current and voltage measurements are taken at a constant interval. These measurements are averaged and the amp-hour rating is the product of the total time and the average current. For the new cell, Duracell claims an average current of .235 amp and a test time of 72 hours, hence the 17 Ah rating. An average voltage of 1.1V was given which indicates about 18.7 Wh at  $0^{\circ}C$  with the  $4.7\Omega$  load.

Unfortunately, battery manufacturers don't seem to give discharge characteristics as a function of temperature and constant power, as is approximately the case for a system like RLAM using DC/DC converters. Figure 33 is part of the data sheet from Duracell for the "D" size alkaline cell. From the lowest set of curves, it appears that with the RLAM load at  $0^{\circ}C$ , a cell will give about 75% of the room temperature rated energy which is better than what we usually assume.

We conducted tests in the lab using some partially discharged RLAM battery packs at room temperature (about  $22^{\circ}C$ ) and  $0^{\circ}C$  using a partial RLAM electronics package and its DC/DC converter. The results are shown in Figures 34a-d & 35a-d. These data are admittedly of limited value in determining the capability of a full voltage battery. However, if the straight portions of the voltage curves are extended, increasing the power curve a corresponding amount in time, we can estimate the energy available from a fresh pack to be about 1250Wh and 1700Wh at  $0^{\circ}C$  and  $22^{\circ}C$  respectively.

We are unsure of the reasons for the difference between these measurements and what we estimate from the Duracell data. We suspect that extrapolation of fresh battery data from our measurements is inaccurate and that our tests should be repeated with fresh packs. We could also account for the losses (small) of Schottky protection diodes.

We do know that a pair of these battery packs ran an RLAM for 8 days and was then down to an average cell voltage of .96V. The RLAM is known to draw about 24 W and these packs were kept warm by their own dissipated power in a well insulated case. (We do not have temperature data from this unit but an experimenter is known to have warmed his hands on these packs while in the field.) These numbers suggest power that is not inconsistent with the Duracell data at  $70^{\circ}$  and the RLAM load.

## C Appendix, AD24 Configuration Code

The routine *AD24conf* is used to configure each AD24 board in a system.

```
***** AD24conf *****
void AD24conf(int iobase, int numchan, unsigned int adc_control_byte)
{
    int i,j,k,p;
    unsigned int m,x;

    p = iobase + CTRL;

    /* state of CTRL port prior to entering this routine is 0x80 */
    /* ADC reset for this card has been applied prior to this routine */
    outport(p,0xfe);      /* HI on R/W is read mode for ADC */
    outport(p,0xfa);      /* set write mode in prep for selection of ch 0 */

    for(i=0;i<numchan;i++)
    {
        /* CTRL port bits: 7   6   5   4   3   2   1   0 */
        /*          ctl  adr2  adr1  adr0  shift  r/w   rsel  sid */
        /* ctl high allows cpu to clk setup info into ADC's */
        /* adr2-adr0 for cpu ADC selection for loading config info */
        /* shft loads serial data into ADC's on falling edge */
        /* r/w selects read (hi) or write (lo) to ADC...lo for config writes */
        /* rsel selects ADC data when hi and status info when lo */

        x = adc_control_byte;
        k = i<<4;
        for(j=0;j<8;j++) /* loop handles all 8 CTRL byte bits */
        {
            /* ADC control word is loaded MSB first, adc_control_byte is: */
            /* PWDN      ORCAL  USEOR  CSEL  N/A  DECc  DECb  DECa */
            /* lo      lo      lo      lo      lo      ADC output rate */

            m = ((x&0x80)>>7); /* msb into lsb */
            outport(p, ((k|0x8a)|m));
            /* serial bit is loaded on falling edge of SHFT */
            outport(p, ((k|0x82)|m));
            x = x<<1;
            /* move next bit into uppermost position of lower byte */
        }
        outport(p,0xfe);      /* HI on R/W is read mode for ADC */
        outport(p,0x8e);      /* select ch 0, read mode */
        outport(p,0x06);
    }
}
```

The routine *ad24seq* is used to load the RAM based sequencer aboard AD24's with code to invoke 2 of the 3 available modes. Mode 0, resulting in fixed point 16 bit data has not been used as it only works when the most 16 significant bits are selected.

```

***** AD24seq *****
/* Routine to load sequencer fifo on AD24 card for 16 bit pfp and
* 24 bit modes. I/O addressing assumes cards in adjacent IO space
* "mode" variable refers to 1 of 3 modes, 16 bit fixed point (0), 24
* bit fixed point (1) & pseudo fp (2).
*
* Sequencer bit layout (9 bits):
*   d8   d7   d6   d5   d4   d3   d2   d1   d0
*   MARKER   SGG   SFF   SEE   LATCH   AUTOGN  ENABLE   CLK   NRESET
*/

```

```

void AD24seq(int iobase, int numchan, int mode)
{
    int i,j,k,p;

    p = iobase + SEQ;
    /* SEQ = 02H
     * if(mode==2)
     *
     * this code is for pseudo-float (autogain) mode
     * for(i=0;i<numchan;i++)
     *
     * if(i==(numchan-1)) k = (i|0x08)<<5;
     * set last ch mark if last
     * else k = i<<5;
     * outport(p,(k|0x0008));
     * outport(p,(k|0x0009));
     * outport(p,(k|0x000b));
     * outport(p,(k|0x0019));
     *
     * for(j=0;j<6;j++)
     *
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0009));
     *
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x000d));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0009));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0009));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x000d));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0009));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0009));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x000d));
     *     outport(p,(k|0x000b));
     *     outport(p,(k|0x0001));
     *     outport(p,(k|0x0003));
     *     outport(p,(k|0x0001));
     *     outport(p,(k|0x0003));
     */
}

```

```

        outport(p,(k|0x0015));

        for(j=0;j<8;j++)
        {
            outport(p,(k|0x0003));
            outport(p,(k|0x0001));
        }
        outport(p,(k|0x0005));
        outport(p,(k|0x0001));
    }
    outport(p,0x0009);
}

else if(mode==1)      /***** 24 bit code *****/
{
    for(i=0;i<numchan;i++)
    {
        if(i==(numchan-1)) k = (i|0x08)<<5;
        /* set last ch mark if last */ 
        else k = i<<5;
        outport(p,(k|0x0008));
        outport(p,(k|0x0009));
        outport(p,(k|0x000b));
        outport(p,(k|0x0019));
        outport(p,(k|0x000b));
        for(j=0;j<6;j++)
        {
            outport(p,(k|0x0001));
            outport(p,(k|0x0003));
        }
        outport(p,(k|0x0005));
        outport(p,(k|0x0003));
        for(j=0;j<7;j++)
        {
            outport(p,(k|0x0001));
            outport(p,(k|0x0003));
        }
        outport(p,(k|0x0005));
        outport(p,(k|0x0003));
        for(j=0;j<7;j++)
        {
            outport(p,(k|0x0001));
            outport(p,(k|0x0003));
        }
        outport(p,(k|0x0001));
        outport(p,(k|0x0001));
        outport(p,(k|0x0001));
        outport(p,(k|0x0005));
        outport(p,(k|0x0001));
    }
    outport(port,0x0009);
}
}

```

The following code has been used to normalize AD24 board output in modes 1 and 2, to volts at the input to the differential amplifiers.

```

/* normalization of AD24 data
 * mode 1 = fixed point 24 bit (3 byte) data
 * mode 2 = pseudo floating point (2 byte) data
 */

ad24_norm(mode,npts,p,q)
int mode;           /* AD24 operating mode, 1=24bit, 2=pfp          */
int npts;           /* the number of samples in the data buffer        */
unsigned int *p;    /* pointer to data buffer                         */
float *q;           /* resultant array of normalized data, mux'd        */
{
    long          val;           /* working long                                     */
    unsigned char *valc;        /* dummy                                         */
    unsigned char *rawc;        /* dummy                                         */

    rawc = p;
    if(mode==1)
/* 24 bit data, even # ch only, 3 - 16 bit words per 2 samples      */

    {
/* val is a long, valc c dummy unsigned char pointer */
        valc = (unsigned char*)&val;
        for(i=0;i<npts;i++)
        {
            if(i%2==0) /* even #'d ch, OK for nch=1 too */
            {
/* hi byte, 1st FIFO word is MSByte of 1st sample -> MSByte of long */
                valc[3] = rawc[1];
/* lo byte, 1st FIFO word is 2MSByte of 1st sample -> 2MSByte of long */
                valc[2] = rawc[0];
/* hi byte, 2nd FIFO word is LSByte of 1st sample -> 3MSByte of long */
                valc[1] = rawc[3];
            }
            else           /* odd # ch                                     */
            {
/* lo byte, 2st FIFO word is MSByte of 2nd sample -> MSByte of long */
                valc[3] = rawc[2];
/* hi byte, 3rd FIFO word is 2MSByte of 2st sample -> 2MSByte of long */
                valc[2] = rawc[5];
/* lo byte, 3rd FIFO word is LSByte of 2st sample -> 3MSByte of long */
                valc[1] = rawc[4];
/* incr ptr to next sample pair                                     */
                rawc += 6;
            }
            valc[0] = 0; /* 0 -> LSByte of long                         */
            q[i] = (double)val*3.352761269e-9;
        }
    }
    else           /* mode 2, pfp                                     */
    {

```

```
    for(i=0;i<npts;i++)
    {
        exp = 1 << ((p[i]&0x03)*3);
        q[i] = (double)(p[i]>>2)*4.5/8192/exp/.625;
        /* The .625 factor is needed because the 24 bit binary value of full */
        /* scale input is 5242880, not 8388608.                                */
    }
}
```

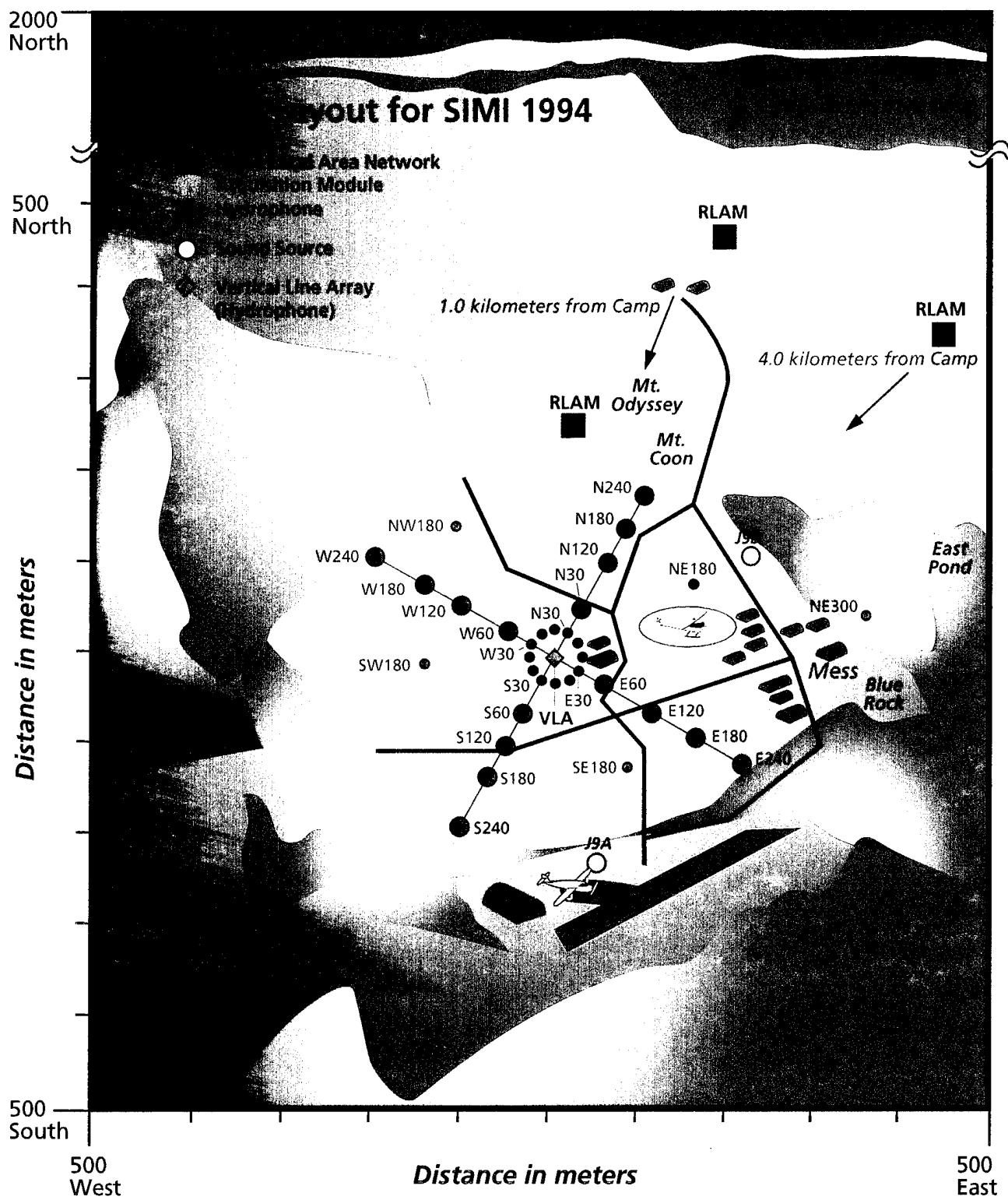


Figure 1. Mock aerial view of the SIMI-94 ice camp. Neither distances or sizes are to scale.

3-AXIS GEOPHONES

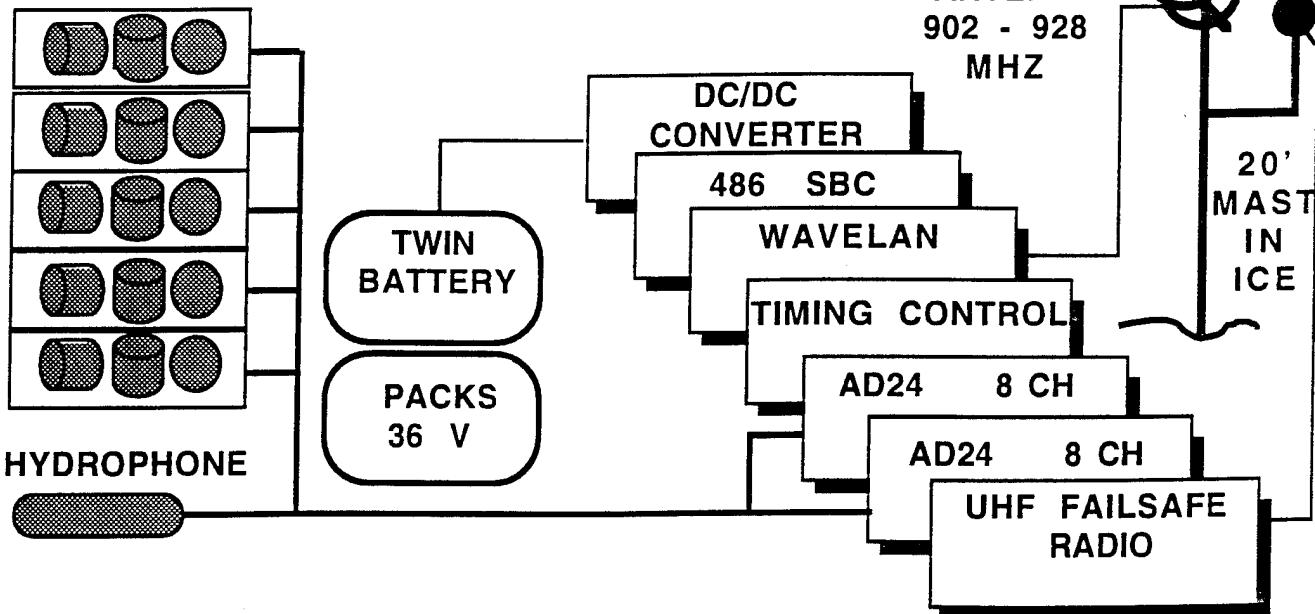


Figure 2. RLAM CONFIGURATION USED FOR SIMI-94 EXPERIMENT

3-AXIS GEOPHONES

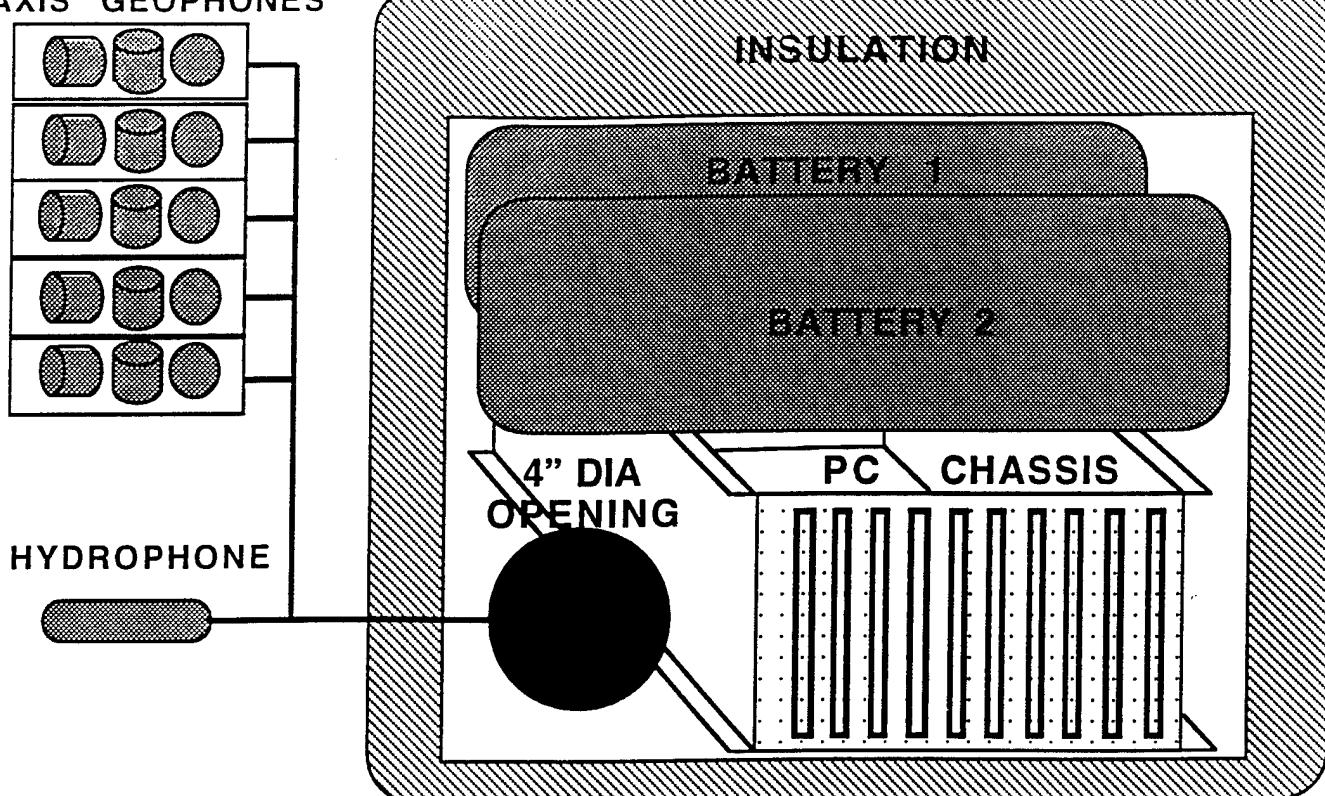


Figure 3. RLAM PACKAGE SHOWING INSULATED CASE, BATTERIES ON TOP OF ELECTRONICS CHASSIS AND SENSOR SUITE.



Figure 4. Photograph showing one of the RLAM deployments during SIMI-94. Note the directional looped YAGI for the 902-928 MHz LAN and the 3 dB omni for the UHF "survival" link.

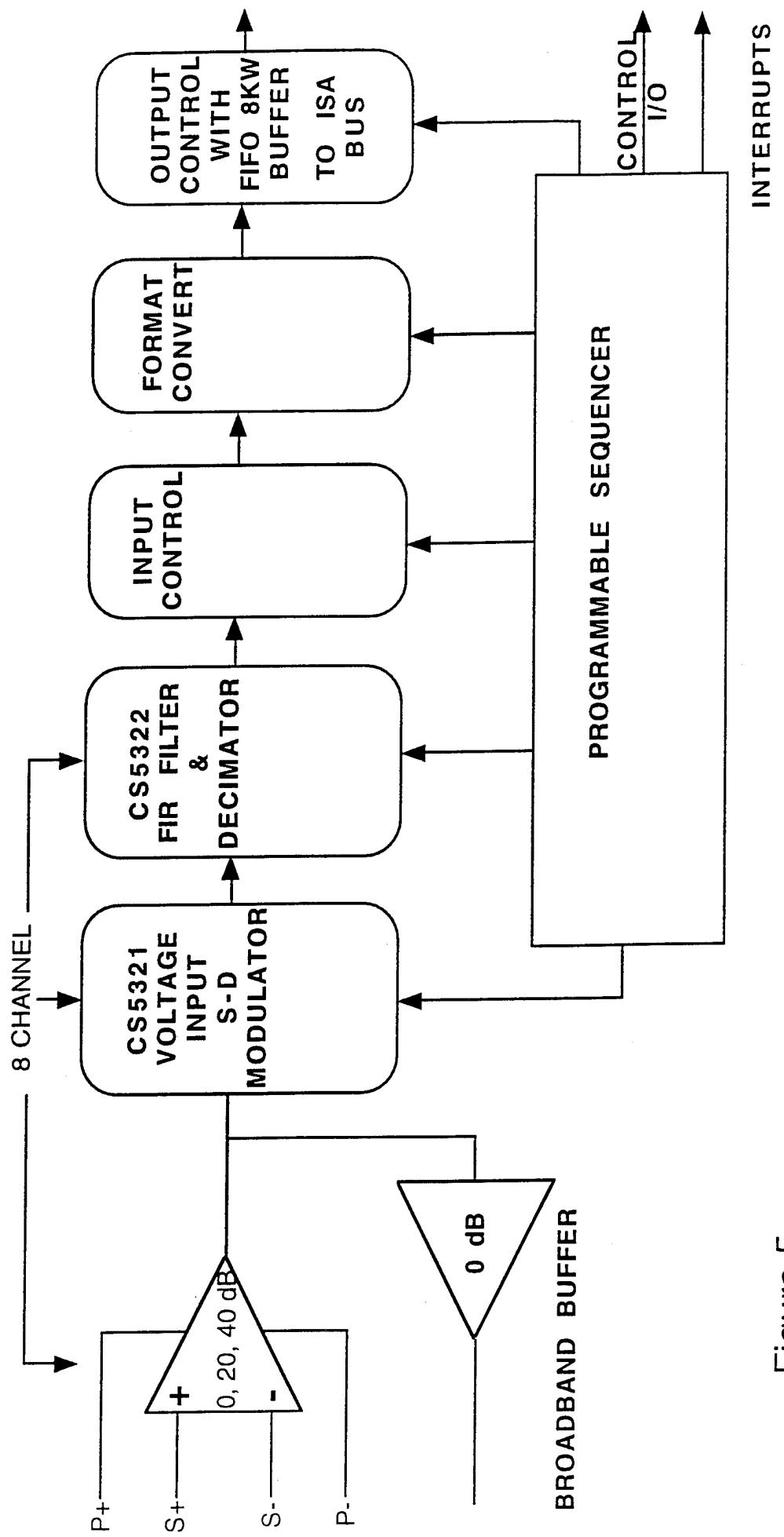
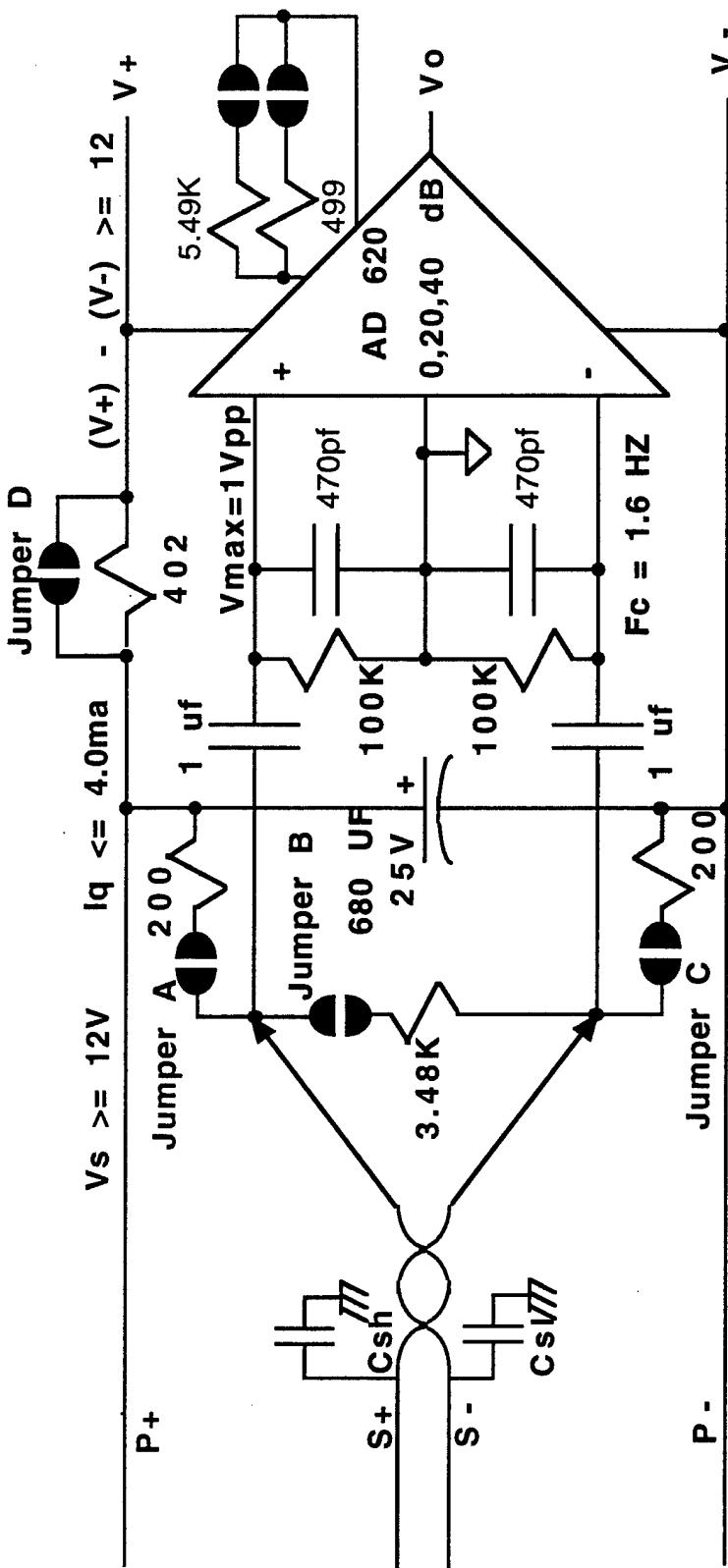


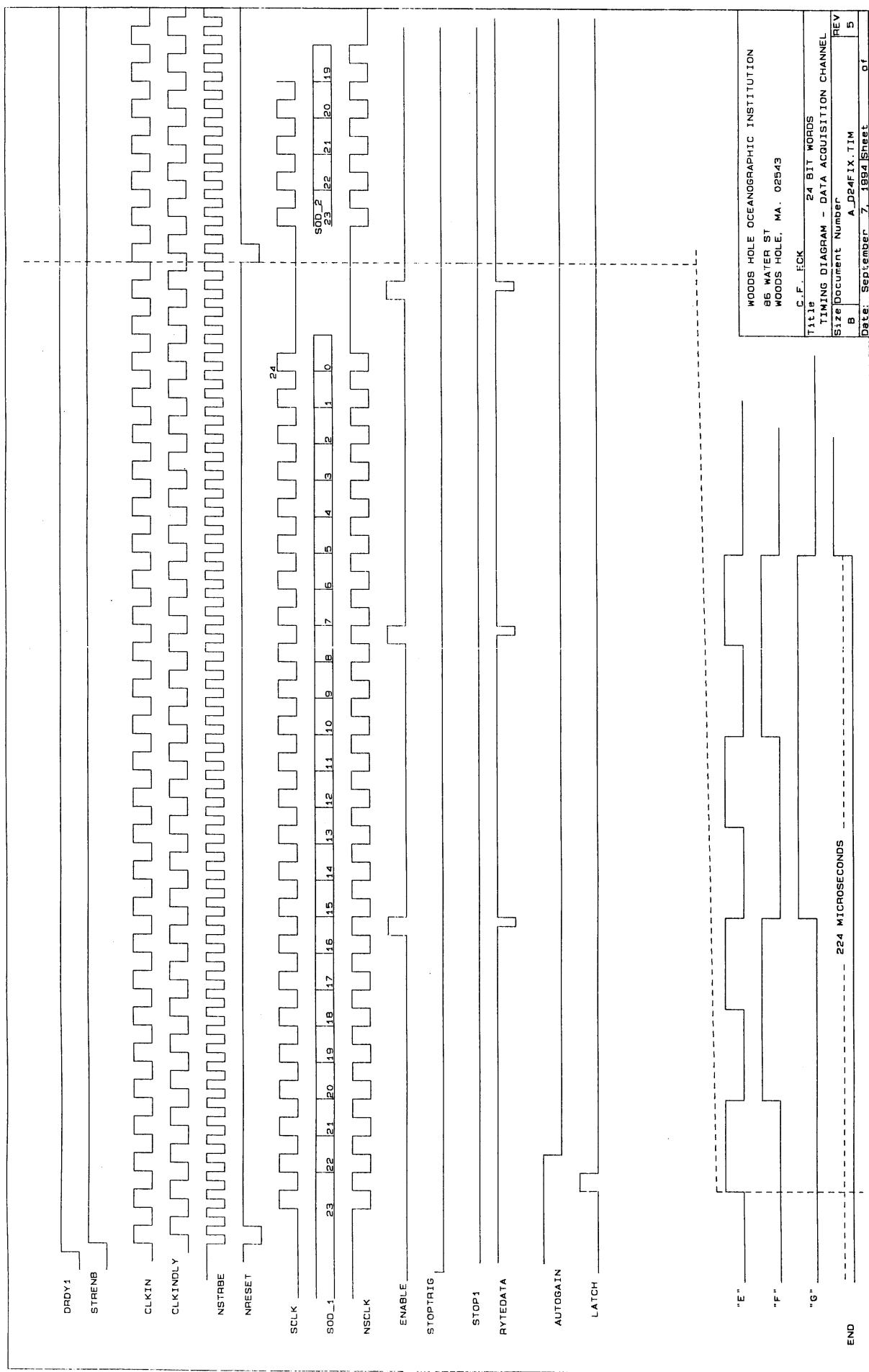
Figure 5. AD24 board functional diagram



**CROSSTALK LEVELS, FROM AVG OF LAB MEASUREMENTS**  
(w/ 600m unshielded cable, 80 pf/m, current mode preamp, stray capacitance as shown, 4 channels)

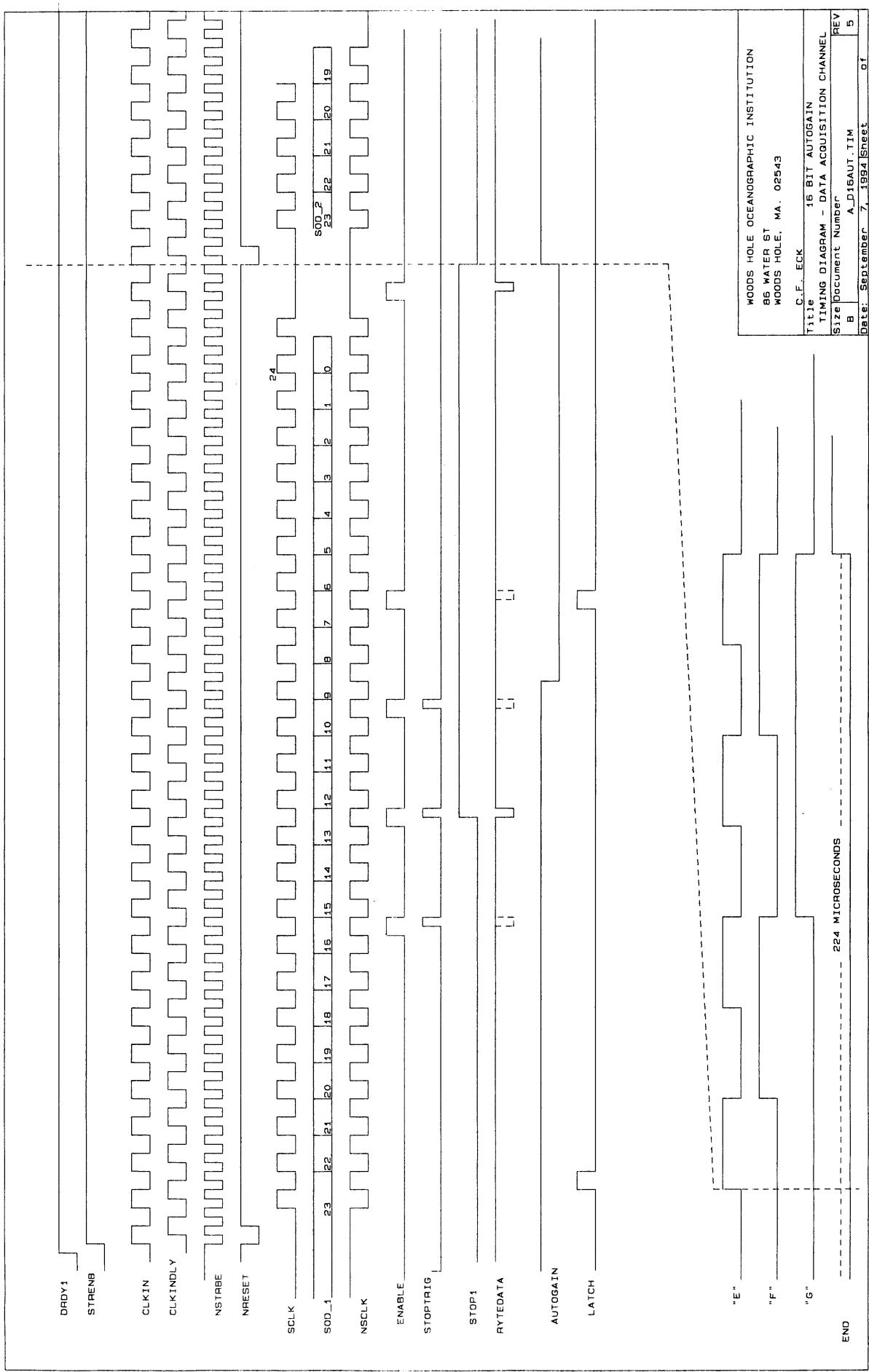
**Figure 6.** Differential signal receiver for the AD24 8 channel, 24 bit ADC board. Jumpers allow 3 different configurations. For voltage mode sensors requiring power, connect jumpers D and B. For voltage mode sensors such as geophones requiring no power, connect jumper B. For current mode sensors, such as powered hydrophones over an STP, connect jumpers A and C. Crosstalk measurements were taken for current mode with "stray" capacitance as shown @ sensor end, (\*15 kHz measurement taken without stray capacitance).

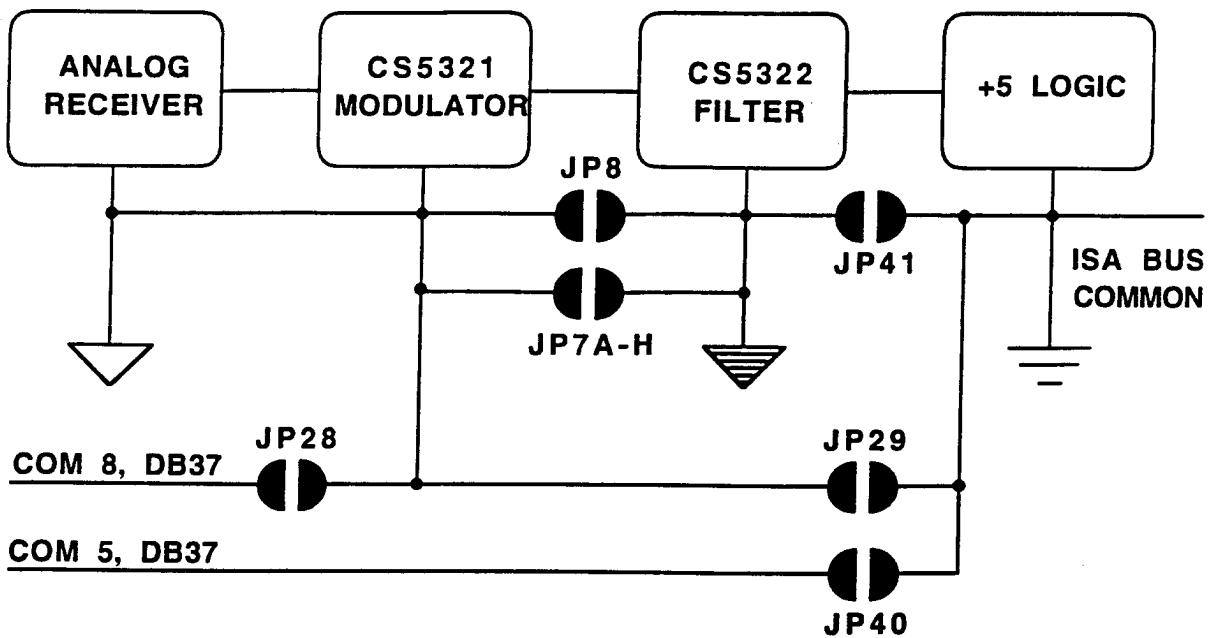
Figure 7. Timing diagram, AD24 control logic for full 24 bit word output, (mode 1).



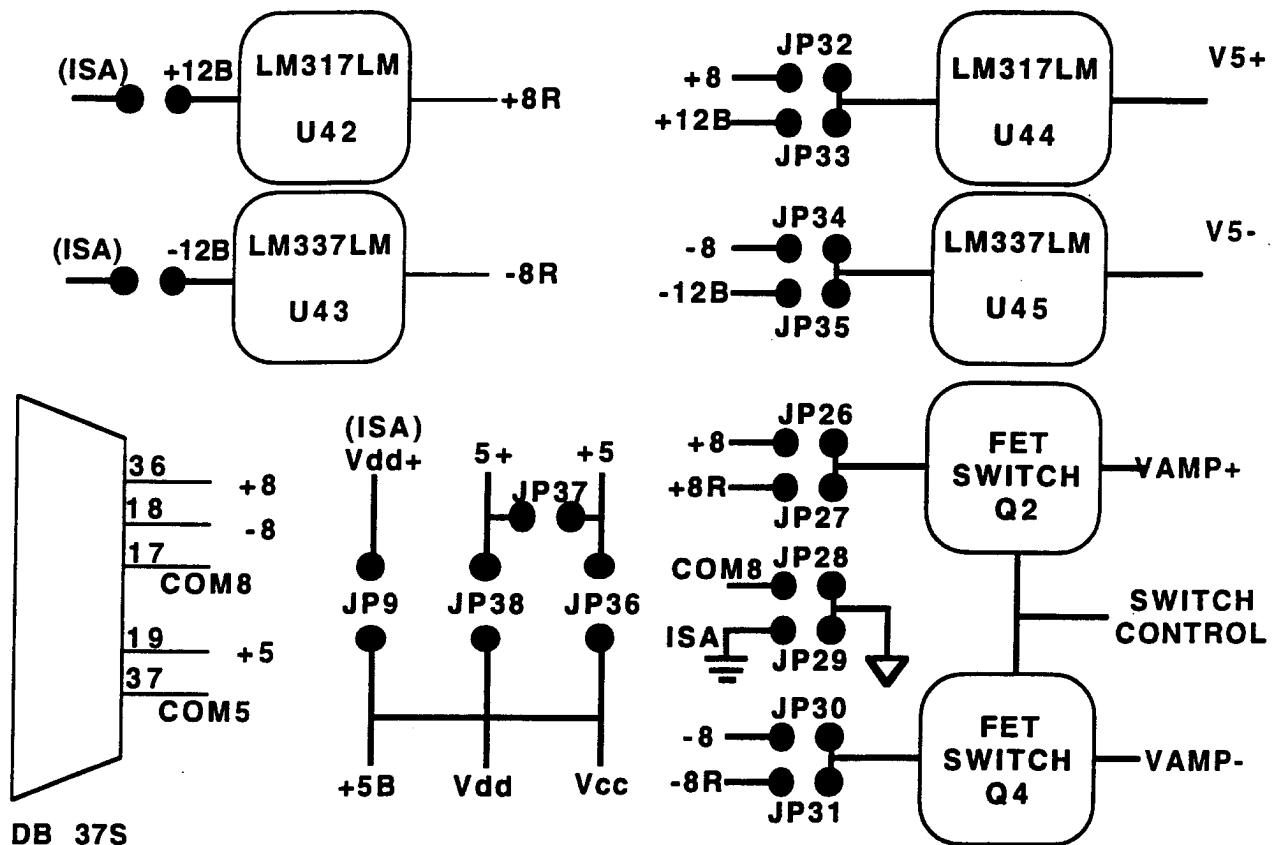
WOODS HOLE OCEANOGRAPHIC INSTITUTION	
86 WATER ST	
WOODS HOLE, MA. 02543	
C. F. ECK	
TIMING DIAGRAM - DATA ACQUISITION CHANNEL	
Size	Document Number
8	A_D24F1X.TIM
Date:	September 7, 1994 Sheet 01
REV	5

Figure 8. Timing diagram, AD24 control logic for 16 bit "pseudo floating point mode, (mode 2).





**Figure 9.** Shown are the AD24 Ground connections, via on-board solder jumpers. We have typically connected JP7A-H, J28, & J41 for RLAM use where all power is taken from the ISA bus +5 & +/-12. A different combination would be used for other power arrangements.



**Figure 10.** AD24 power source jumpers. V5+ and V5- power the ADC's. VAMP+ and VAMP- power the analog receivers. Vcc, Vdd and "+" power digital filter and logic components.

REC STARTED ~ 894  
110 893.91 $f_s = 1953.125$ 

## Recs110 to 111

Box 0 (2)

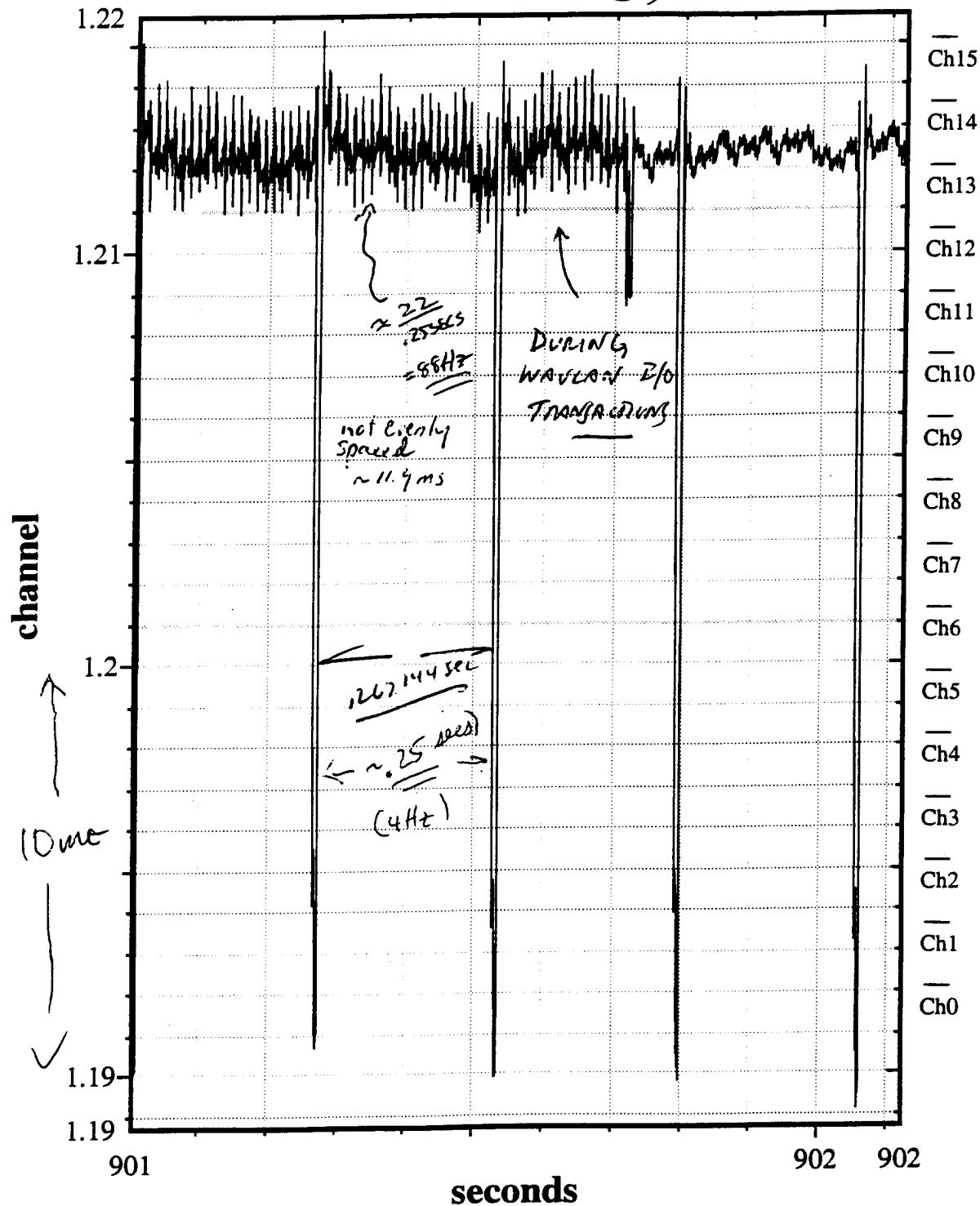
~~8.257555 S/R~~  
8.125464 S/R

Figure 11. An example of an RLAM data trace plot acquired during a deployment during SIMI-94 before the I/O noise problem had been resolved. Note the characteristic large 25 mvpp negative going spikes that coincided with AD24 data FIFO read operations. The smaller spikes between the large correspond to I/O operations associated with radio LAN transmissions.

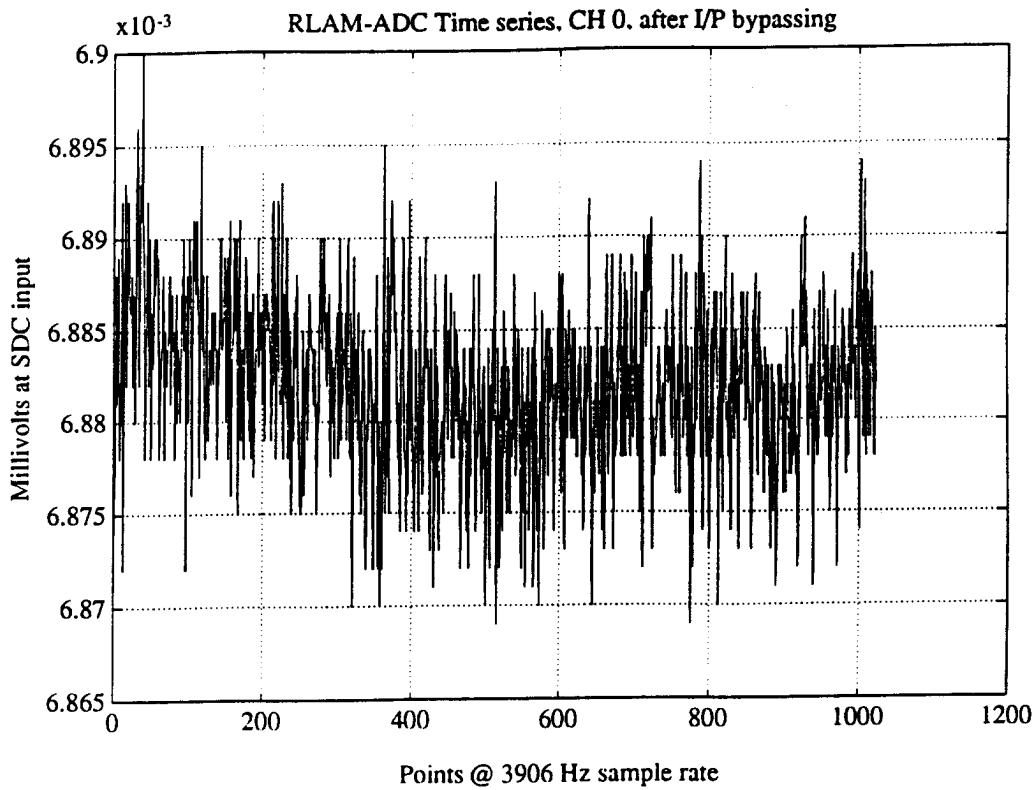


Figure 13. The same RLAM channel as in Figure 12 after bypass capacitors were added. Most channels are similar though a few show a very much reduced replica of Figure 12.

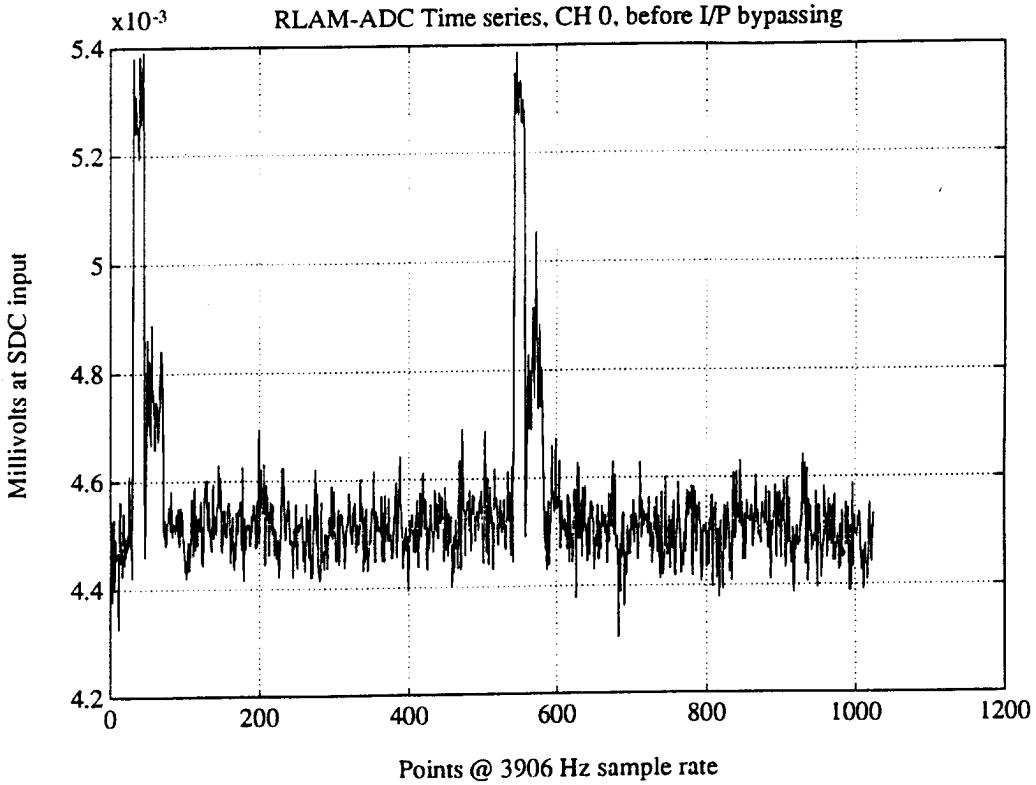
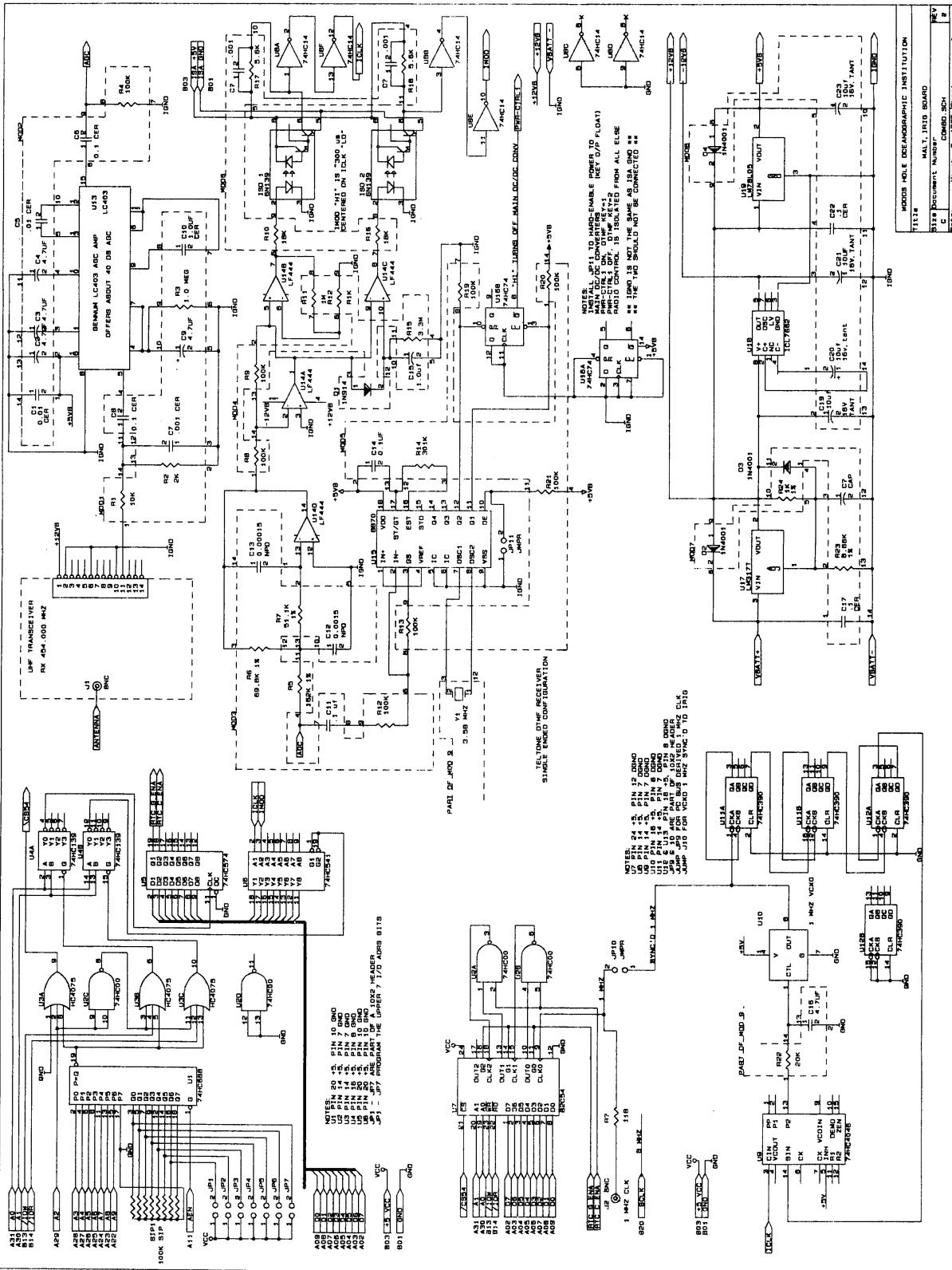
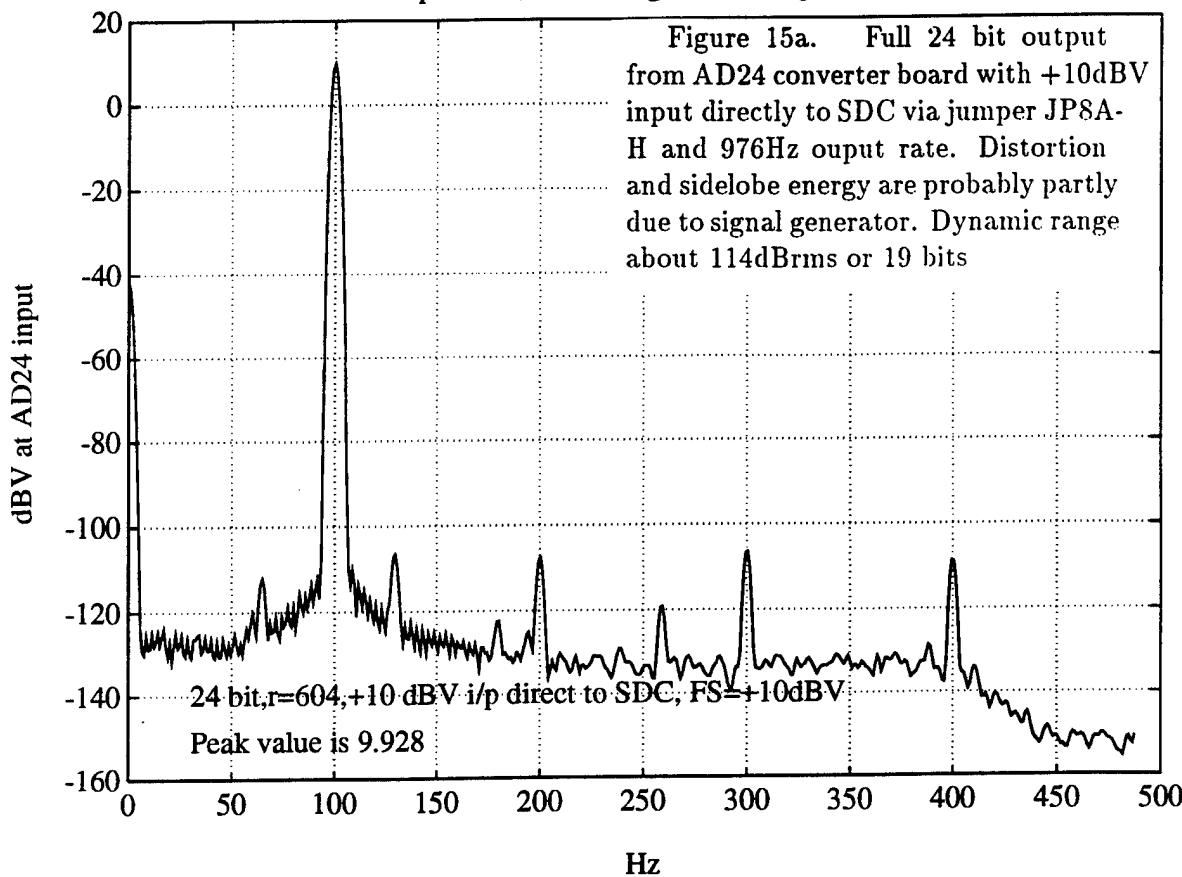


Figure 12. RLAM channel time series before addition of 470pf NPO bypass capacitors in close proximity to the differential amplifier inputs on the AD24 board. A smaller amplitude pulse following the first corresponds to the FIFO readout of the second AD24 board.

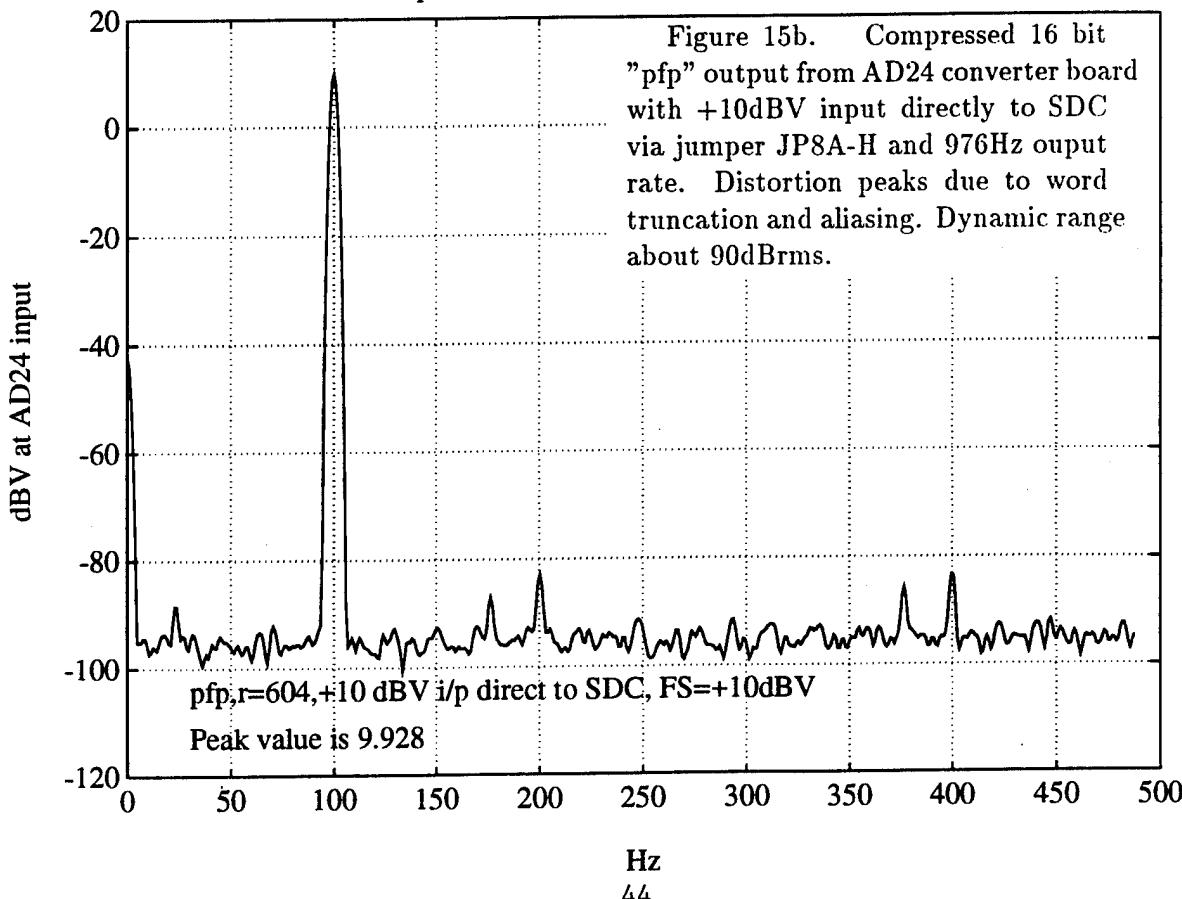
Figure 14. Circuit schematic of RRAM timing control board. Functions included are real-time clock (82C54), DTMF decoder, TRIG reader with AGC, 1 MHz VCO, DC/DC converter shutdown control, UHF radio power regulator.



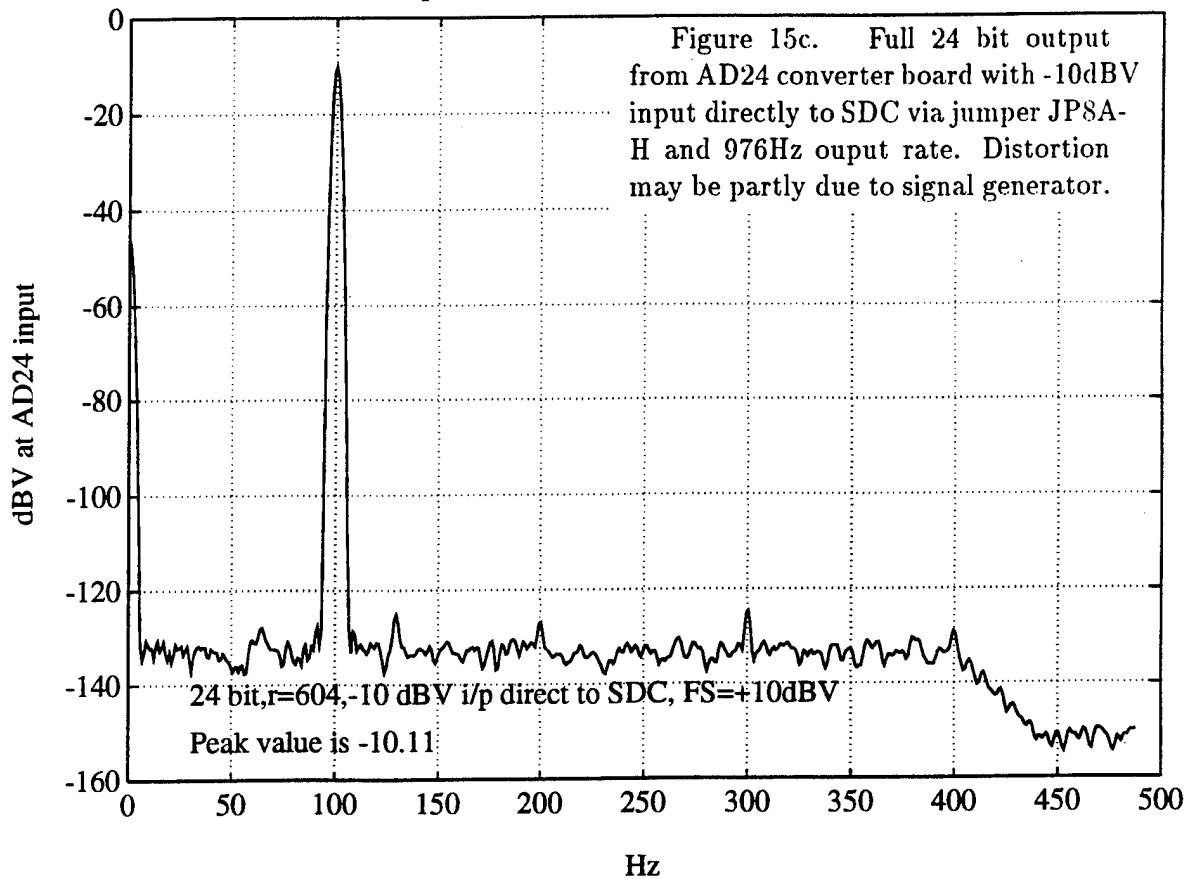
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



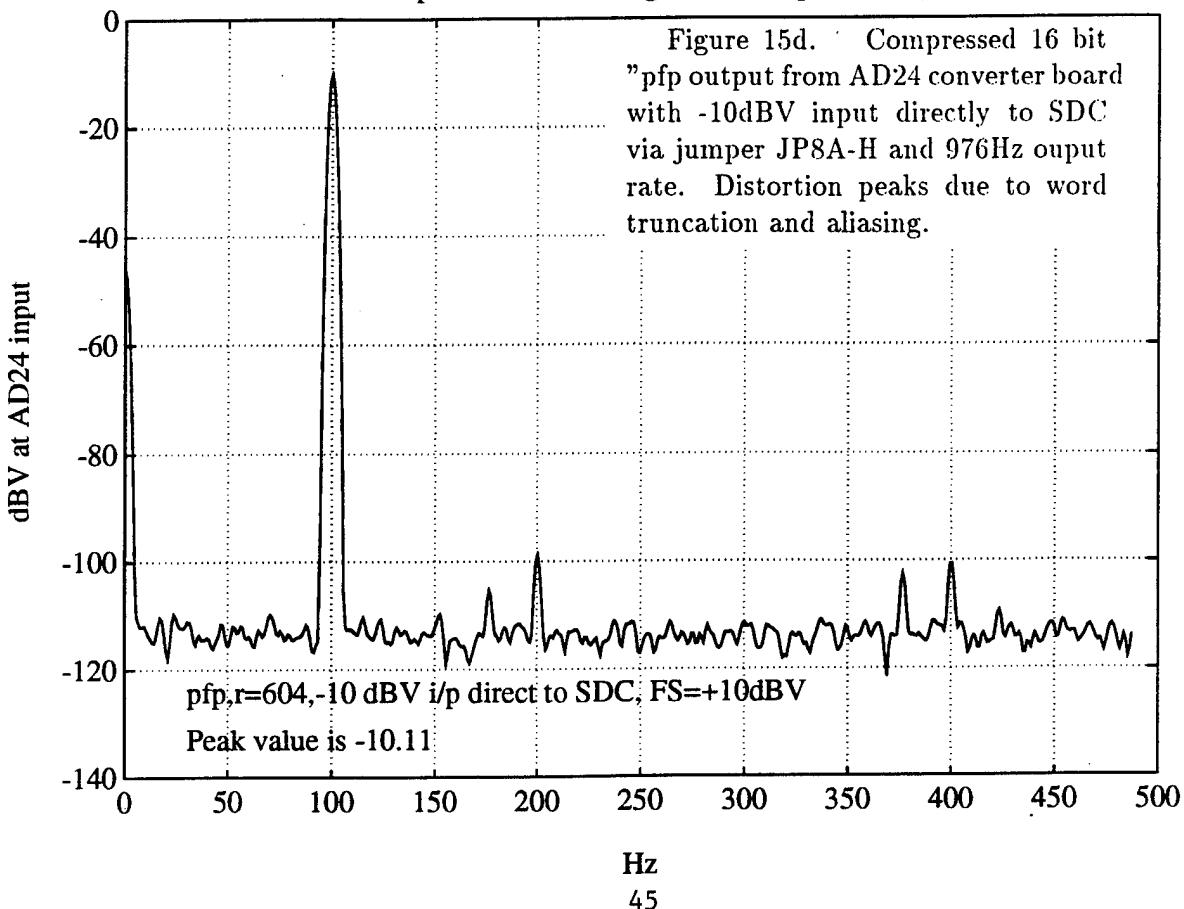
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



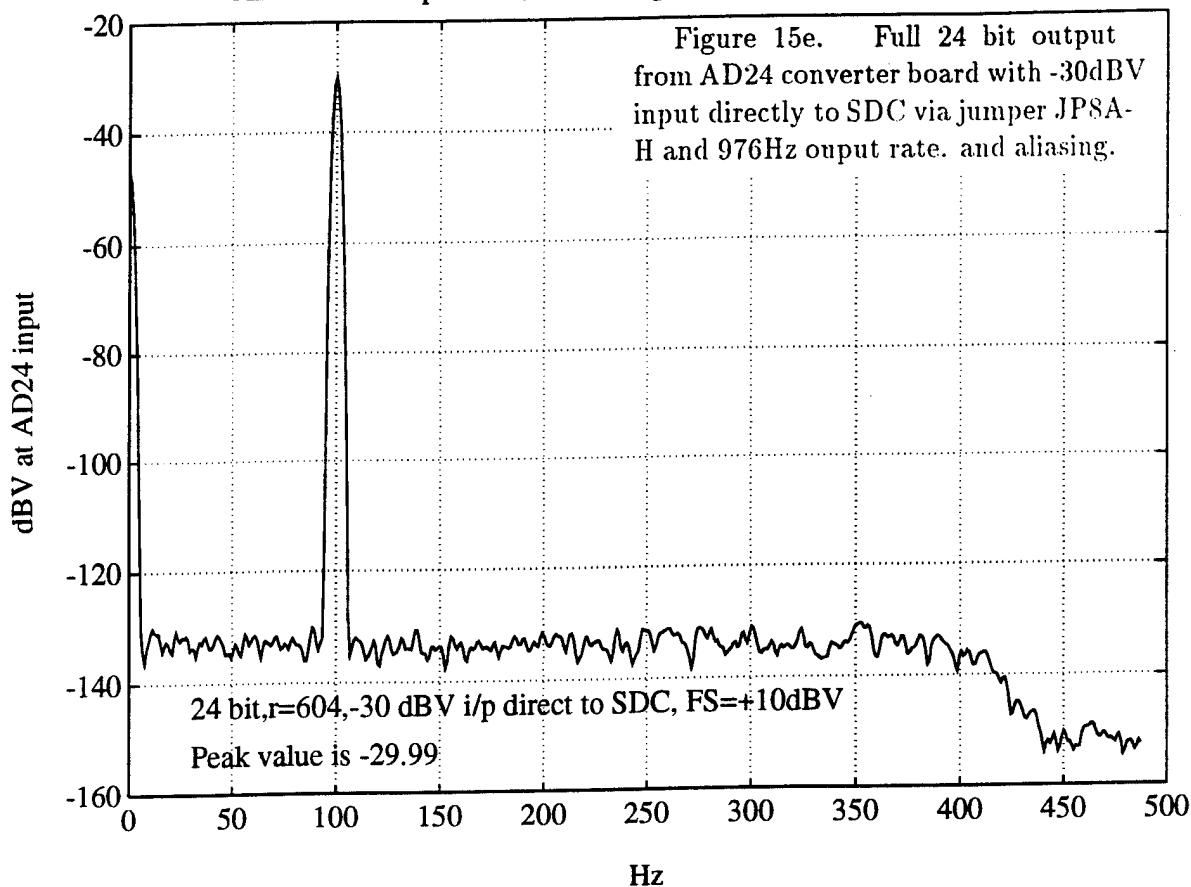
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



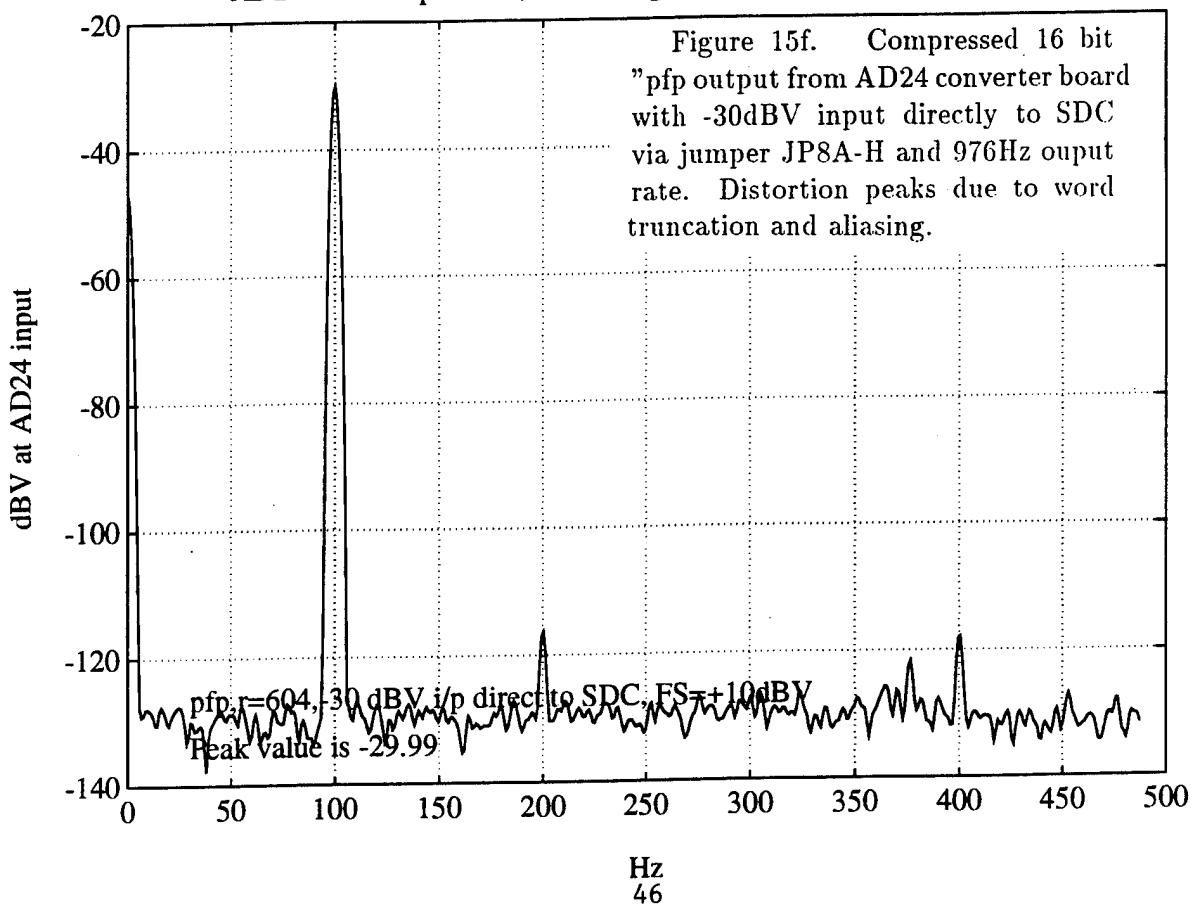
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



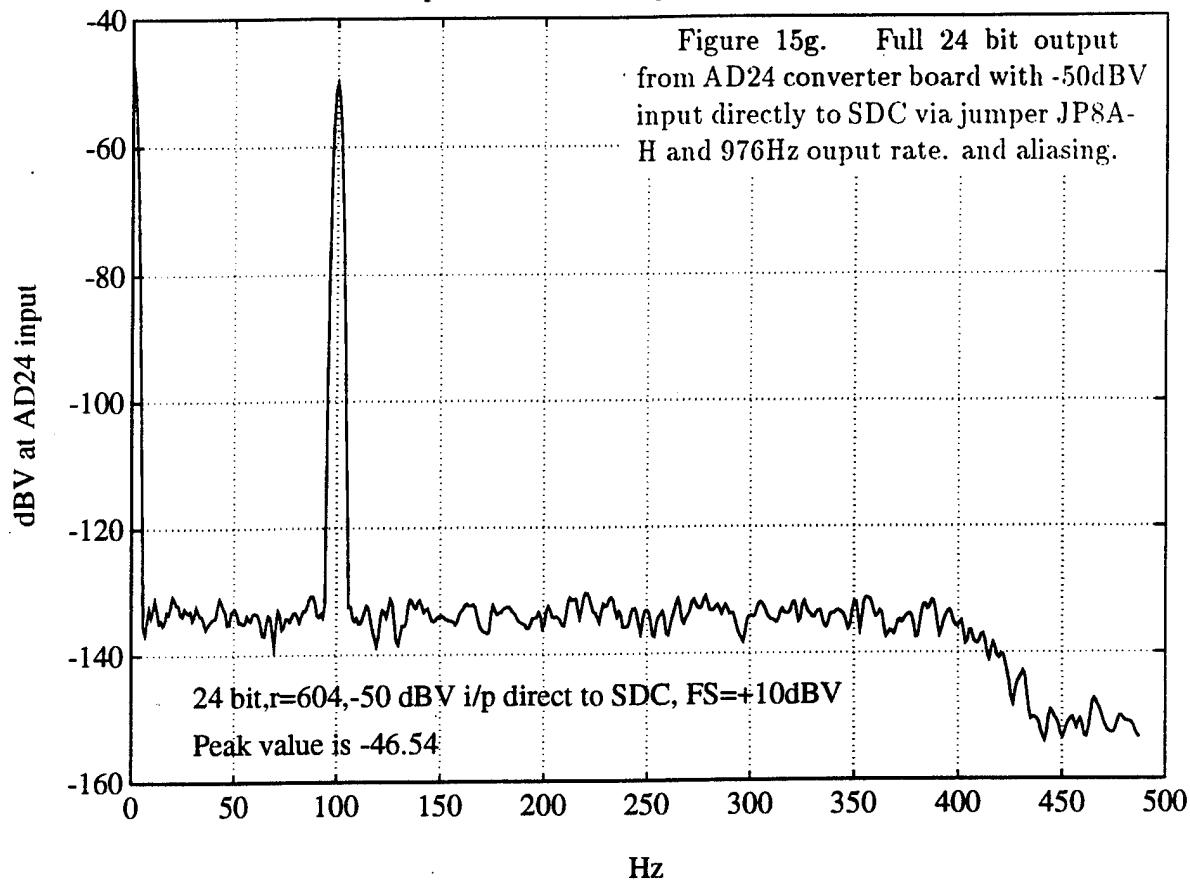
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



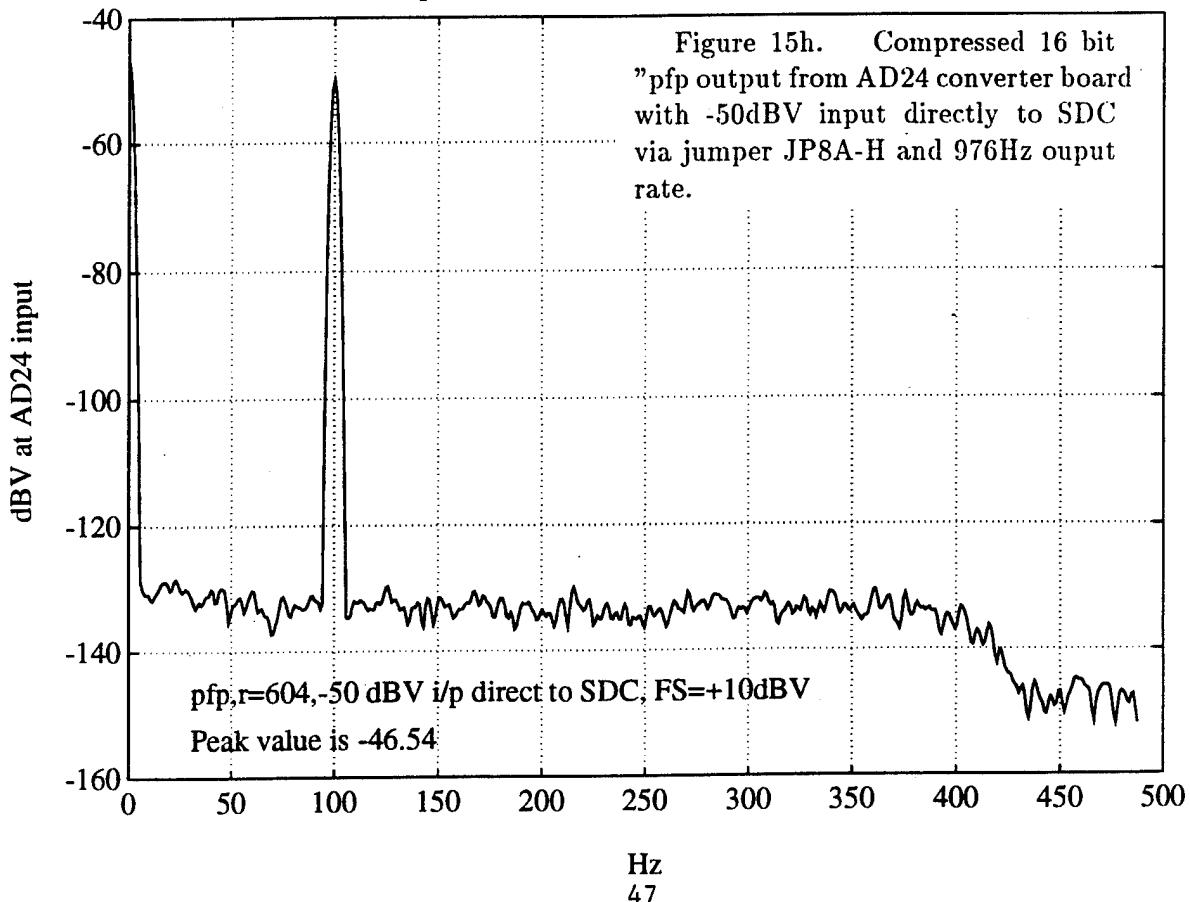
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



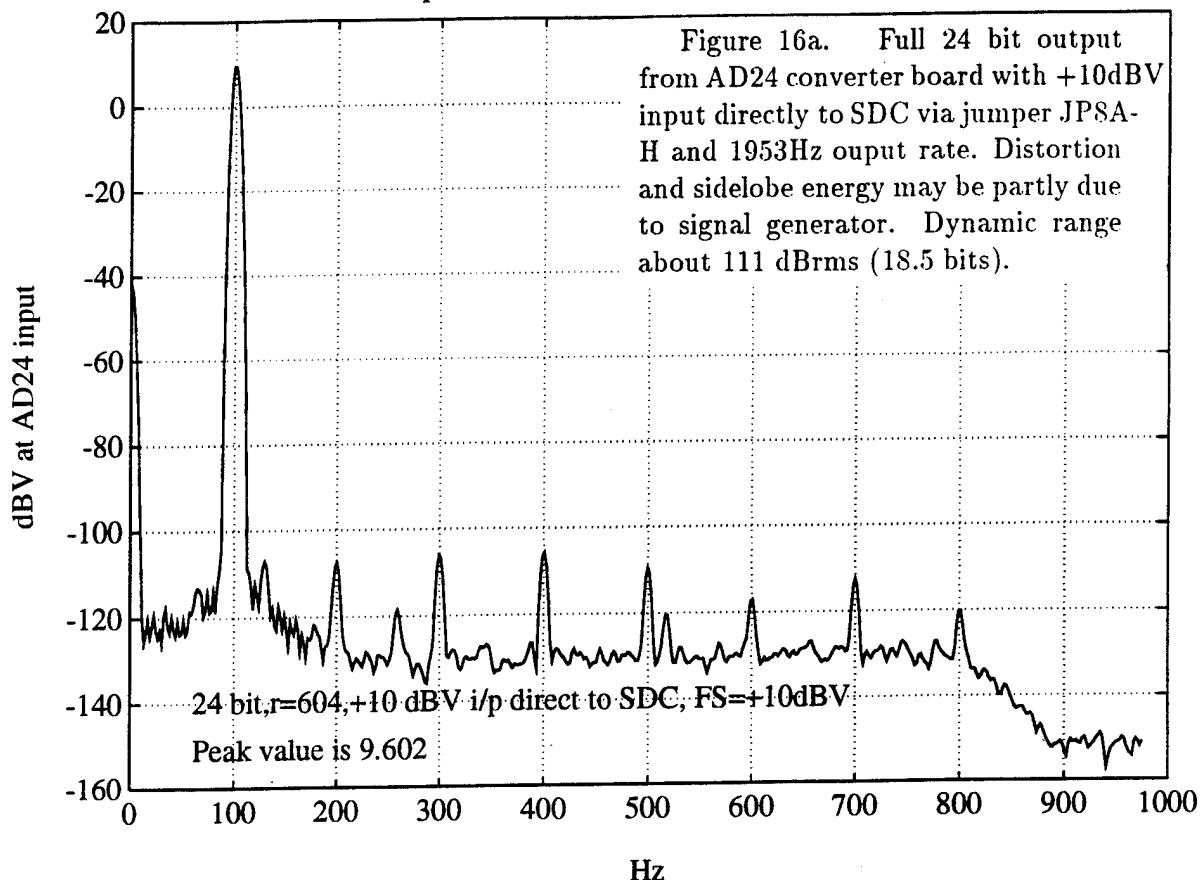
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



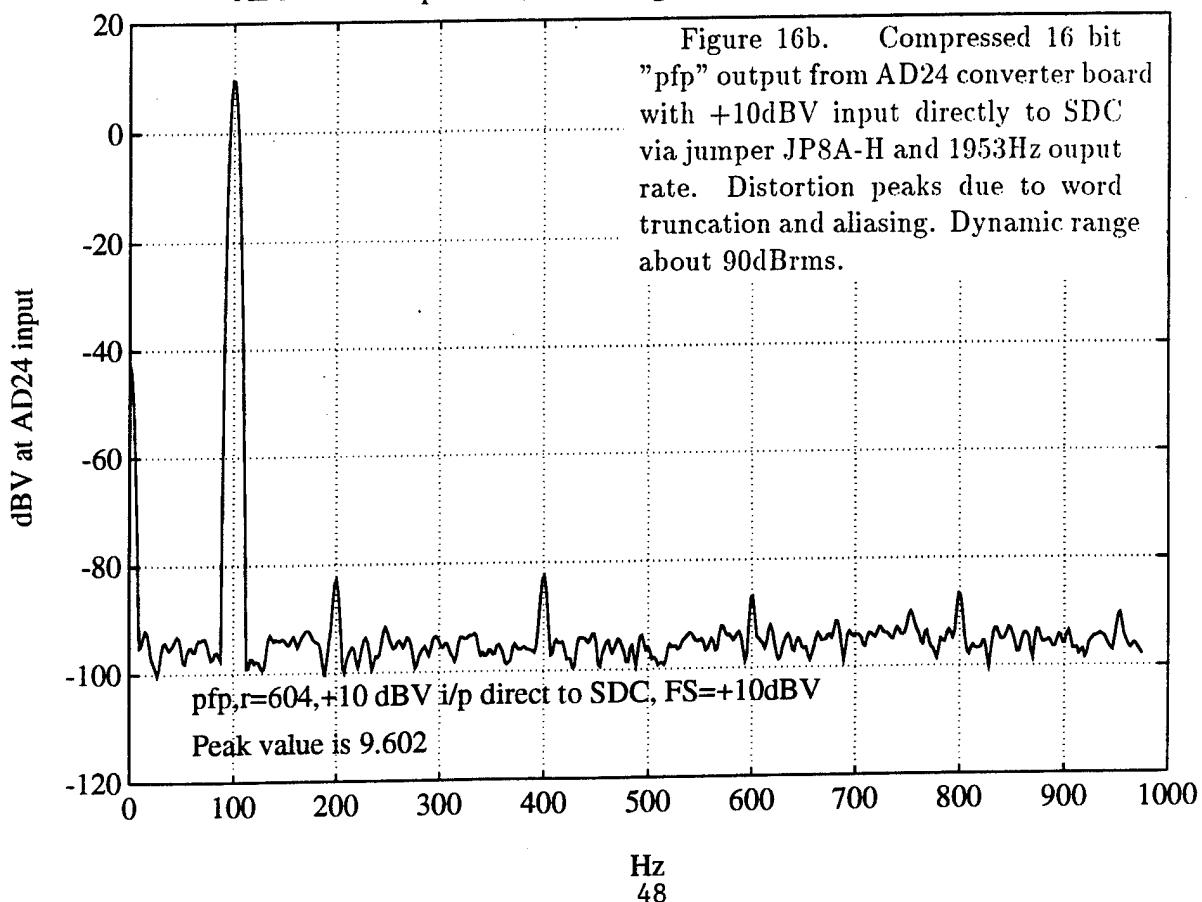
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



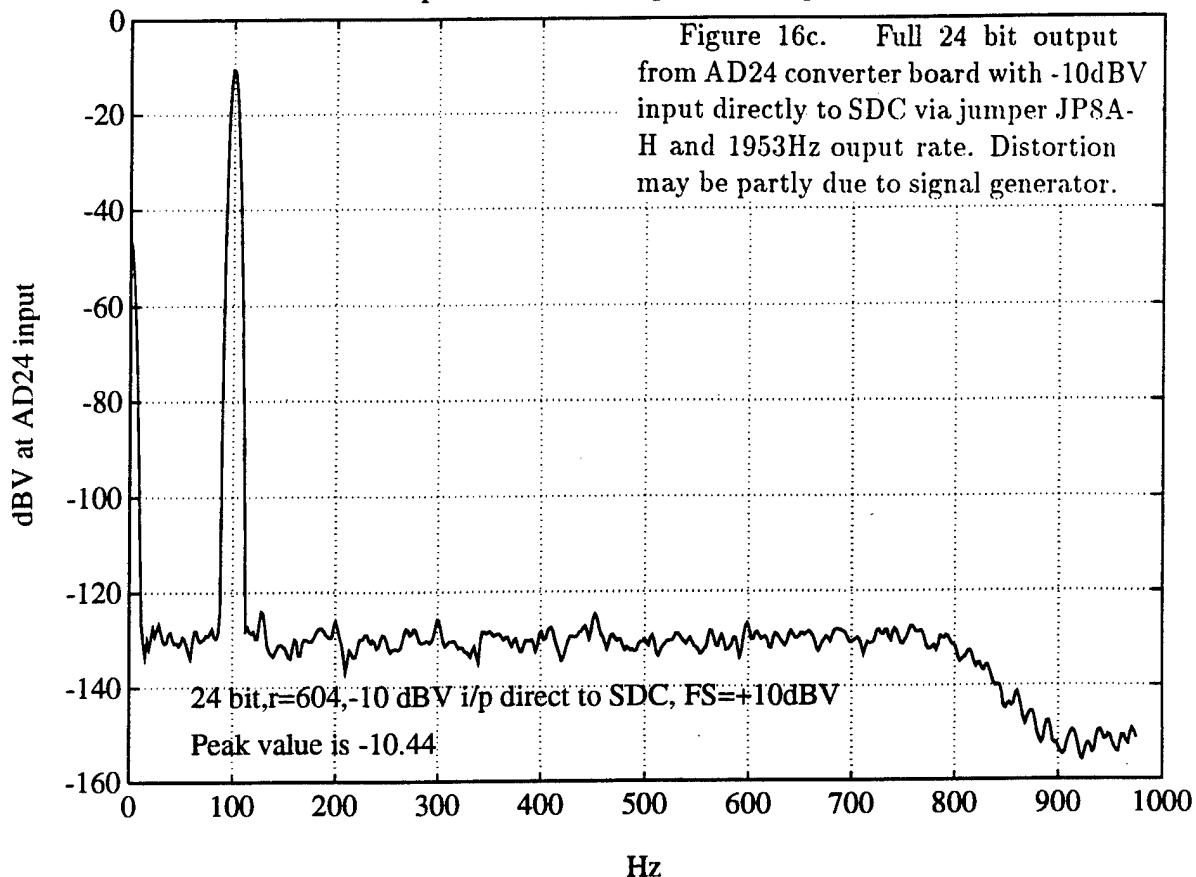
### AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



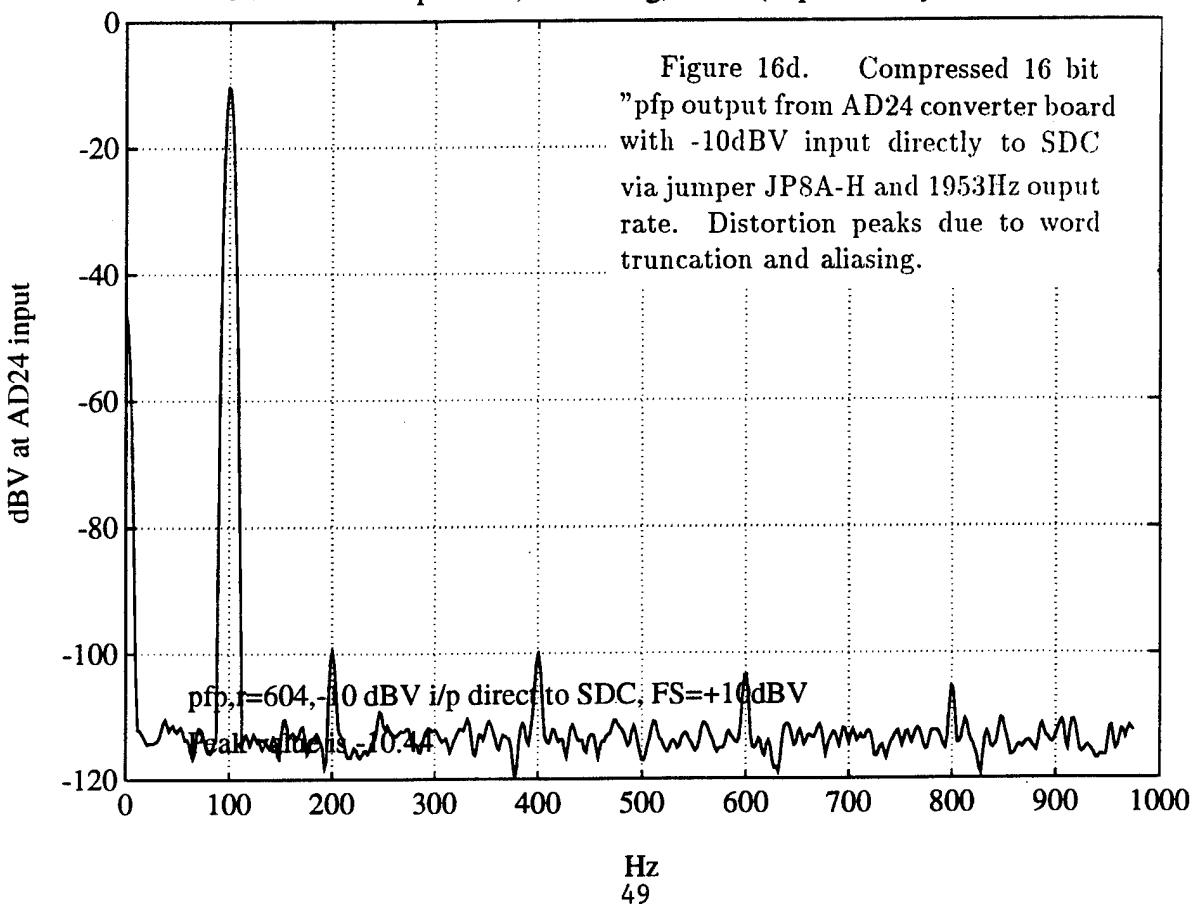
### AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



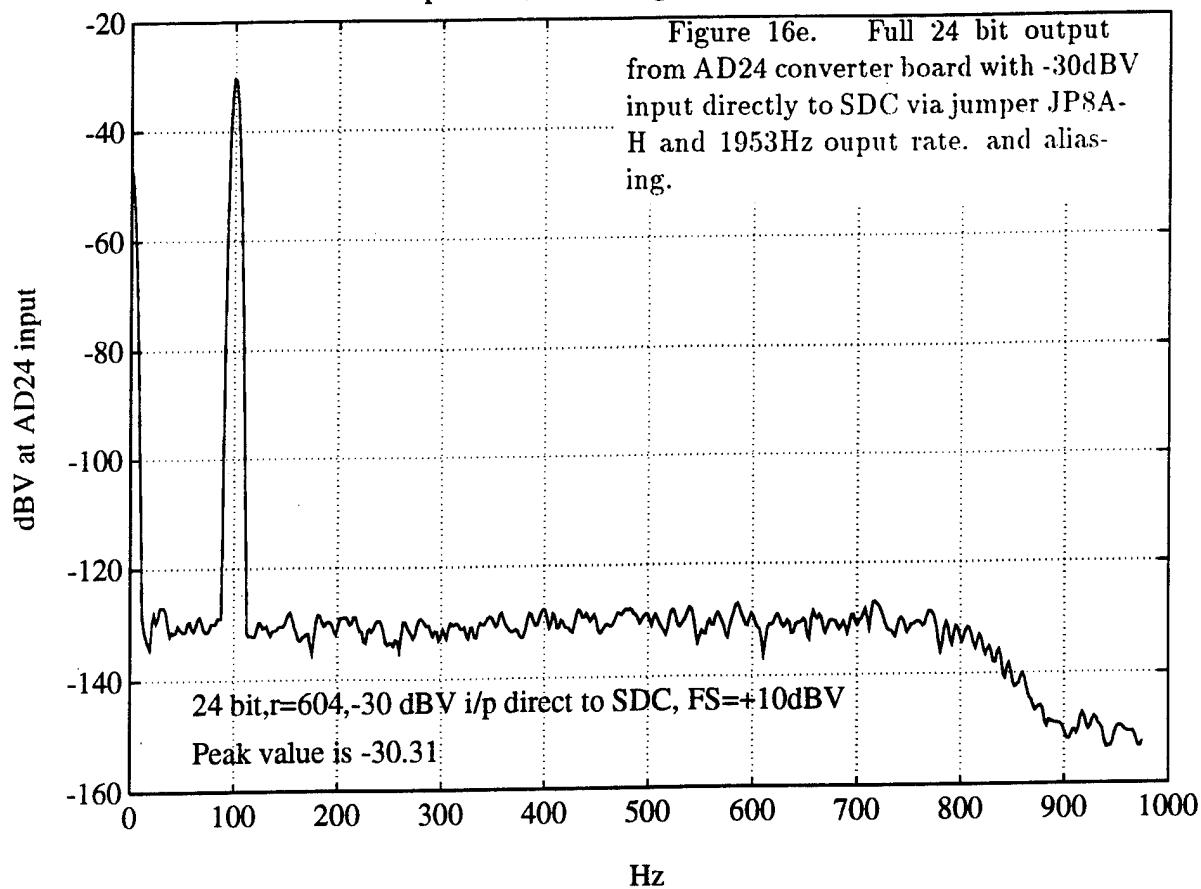
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



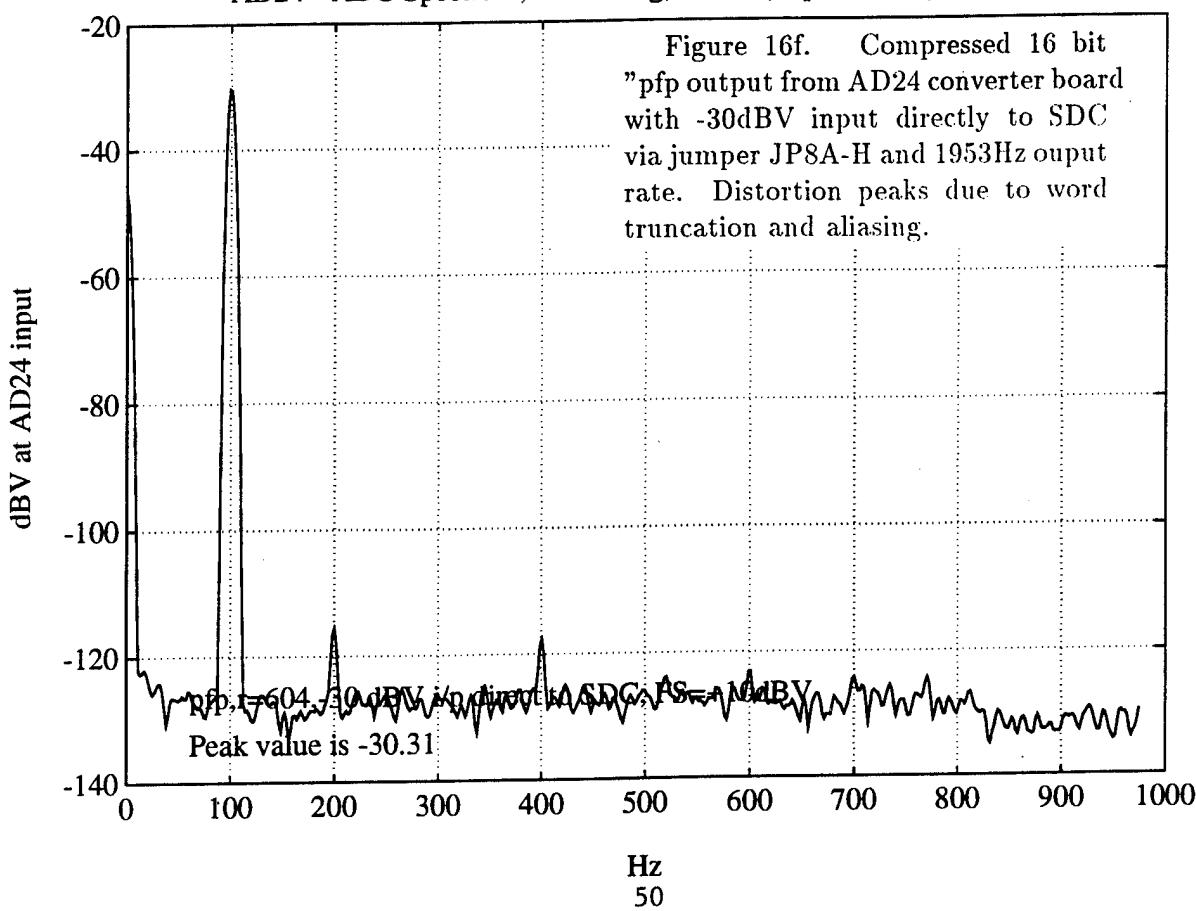
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



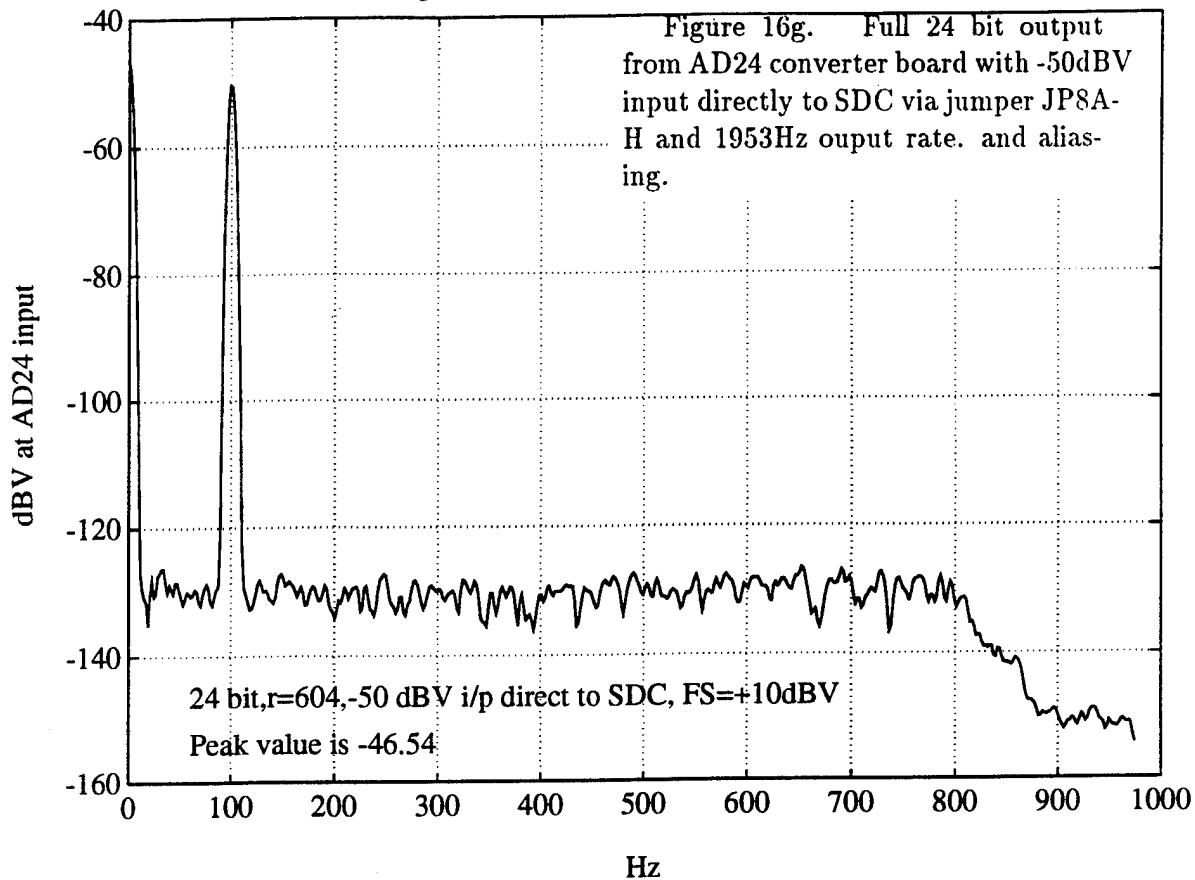
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



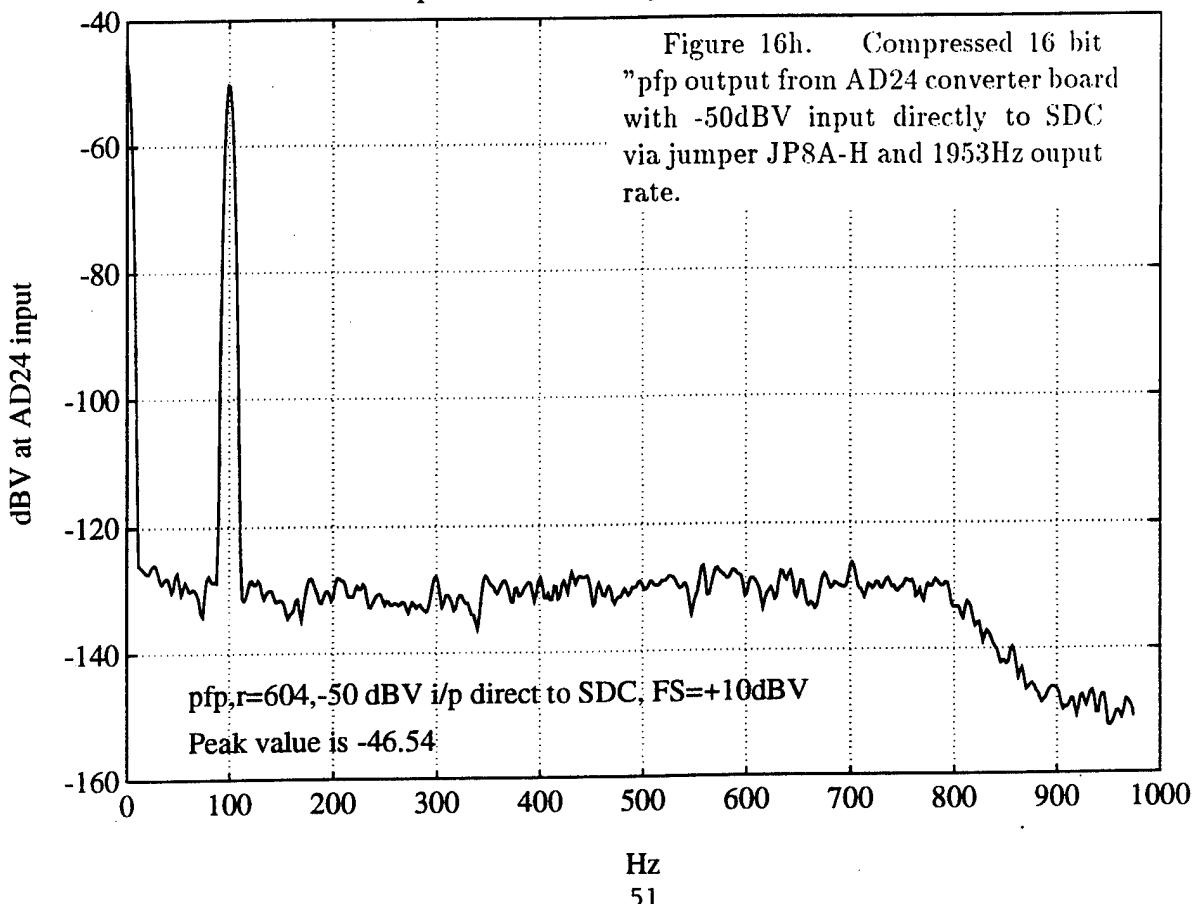
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



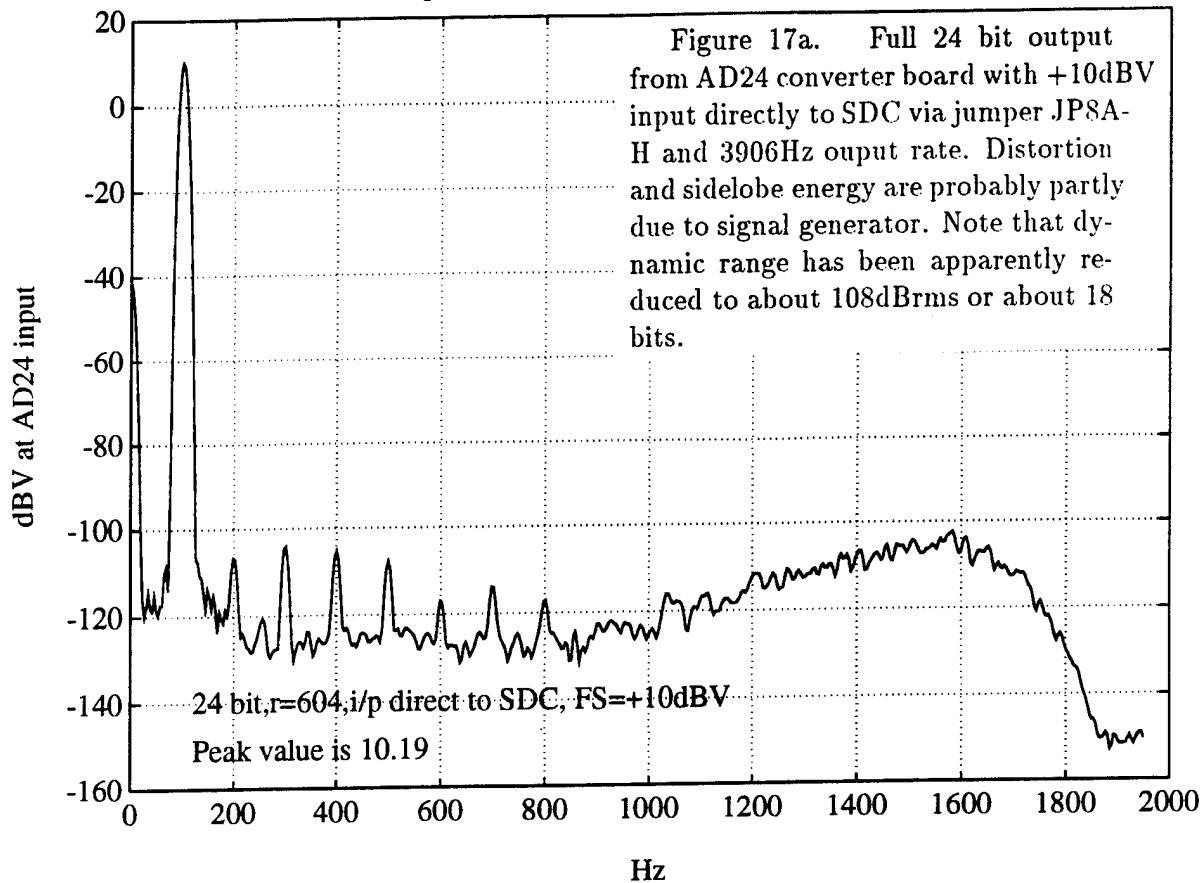
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



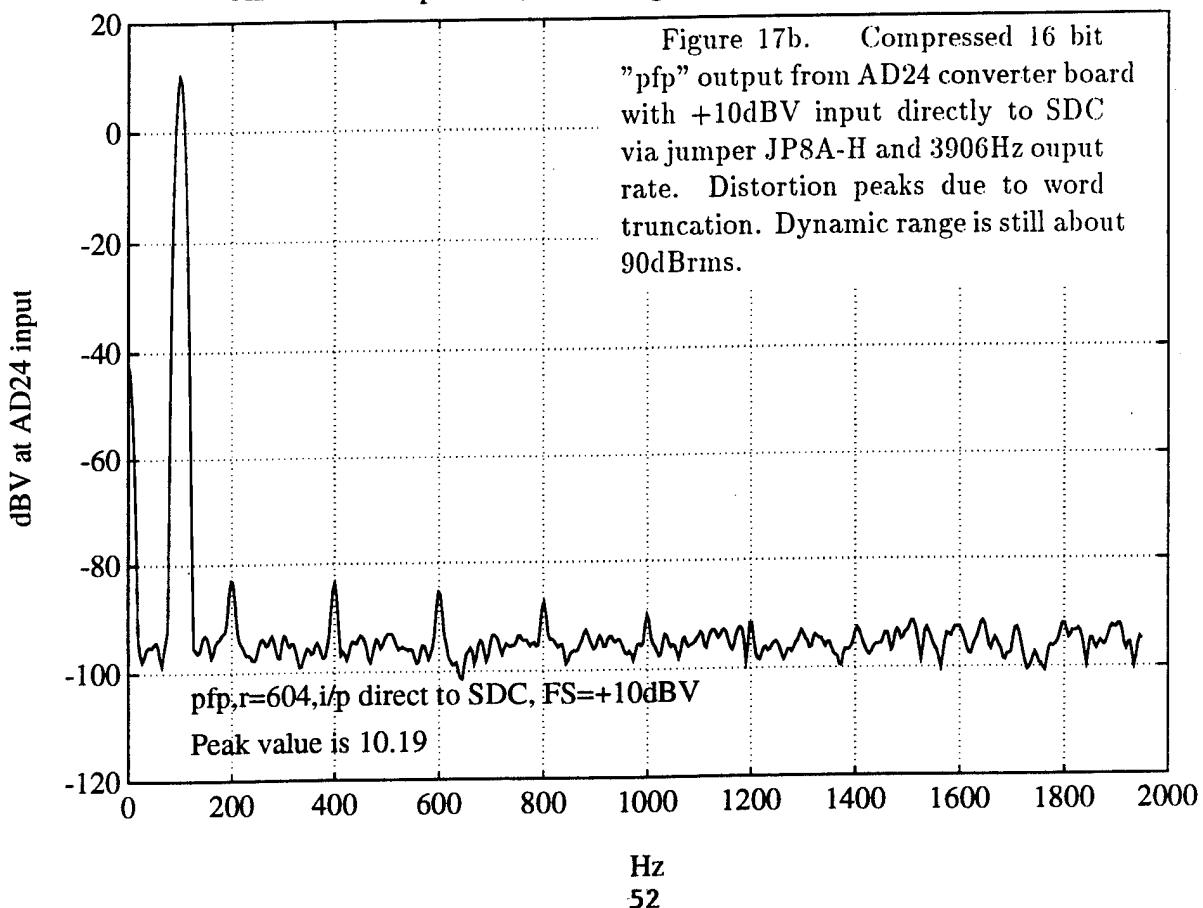
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



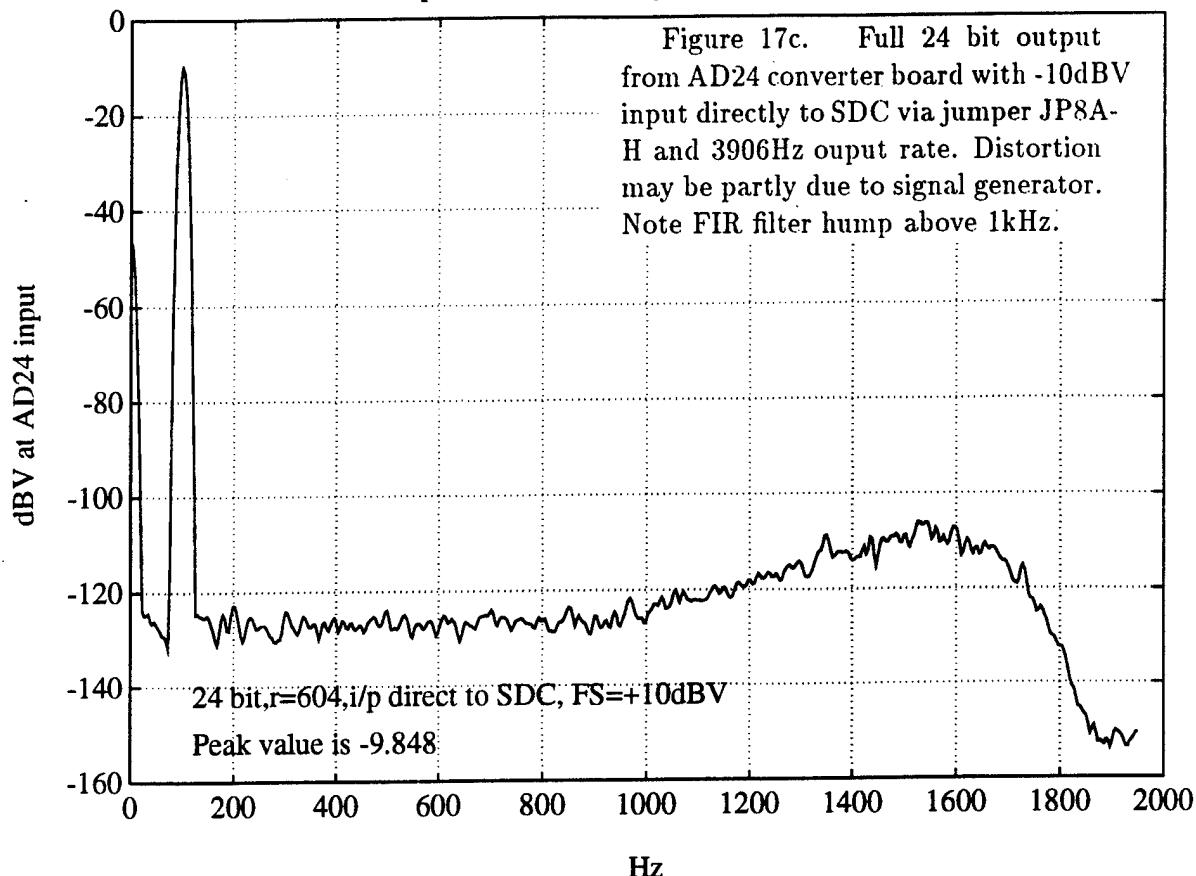
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



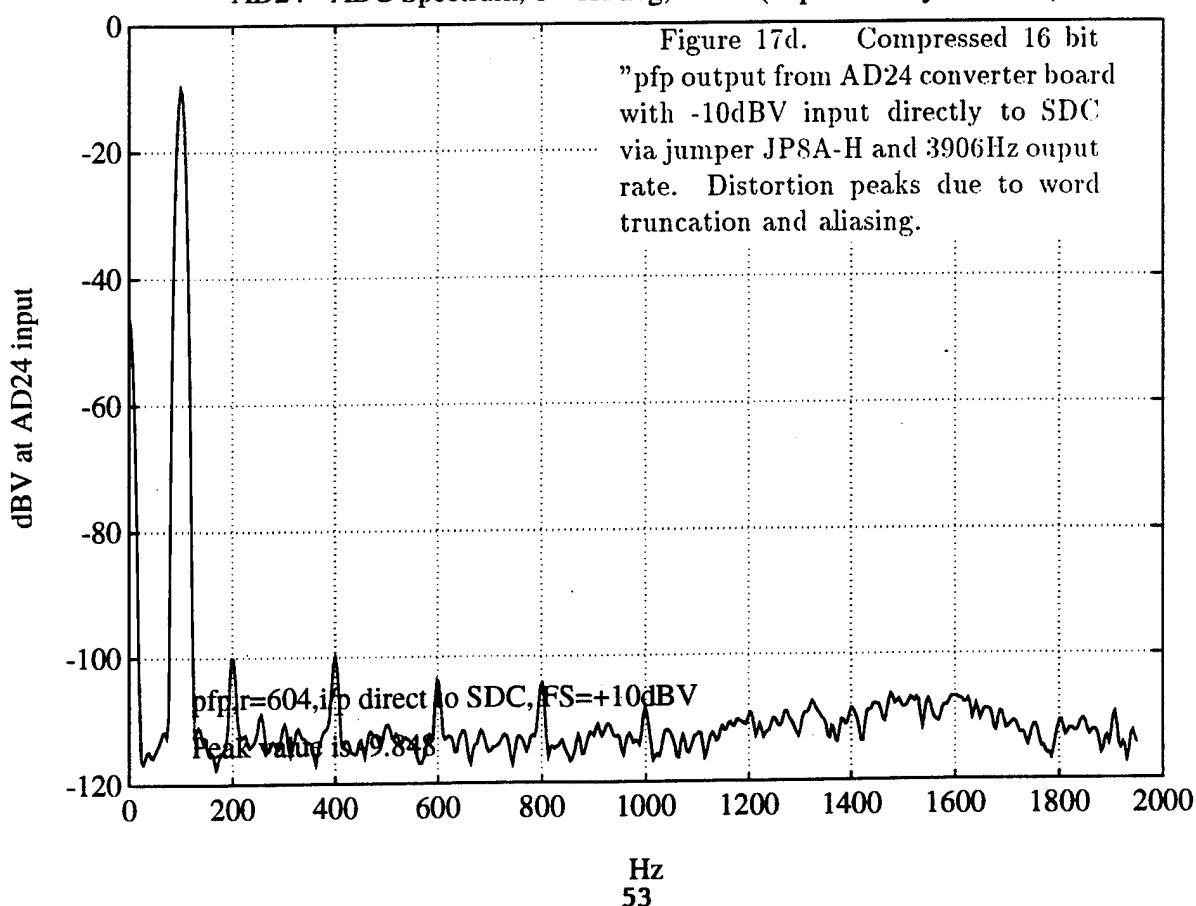
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



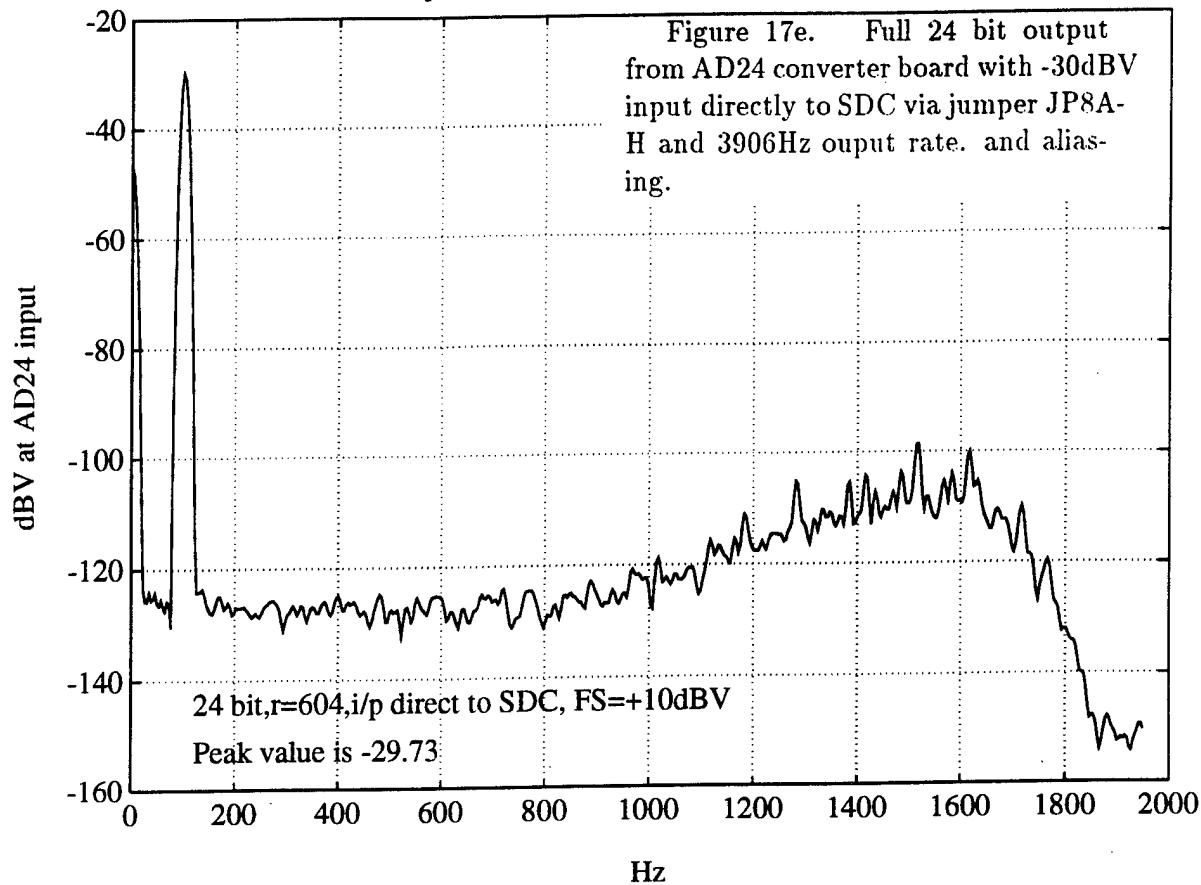
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



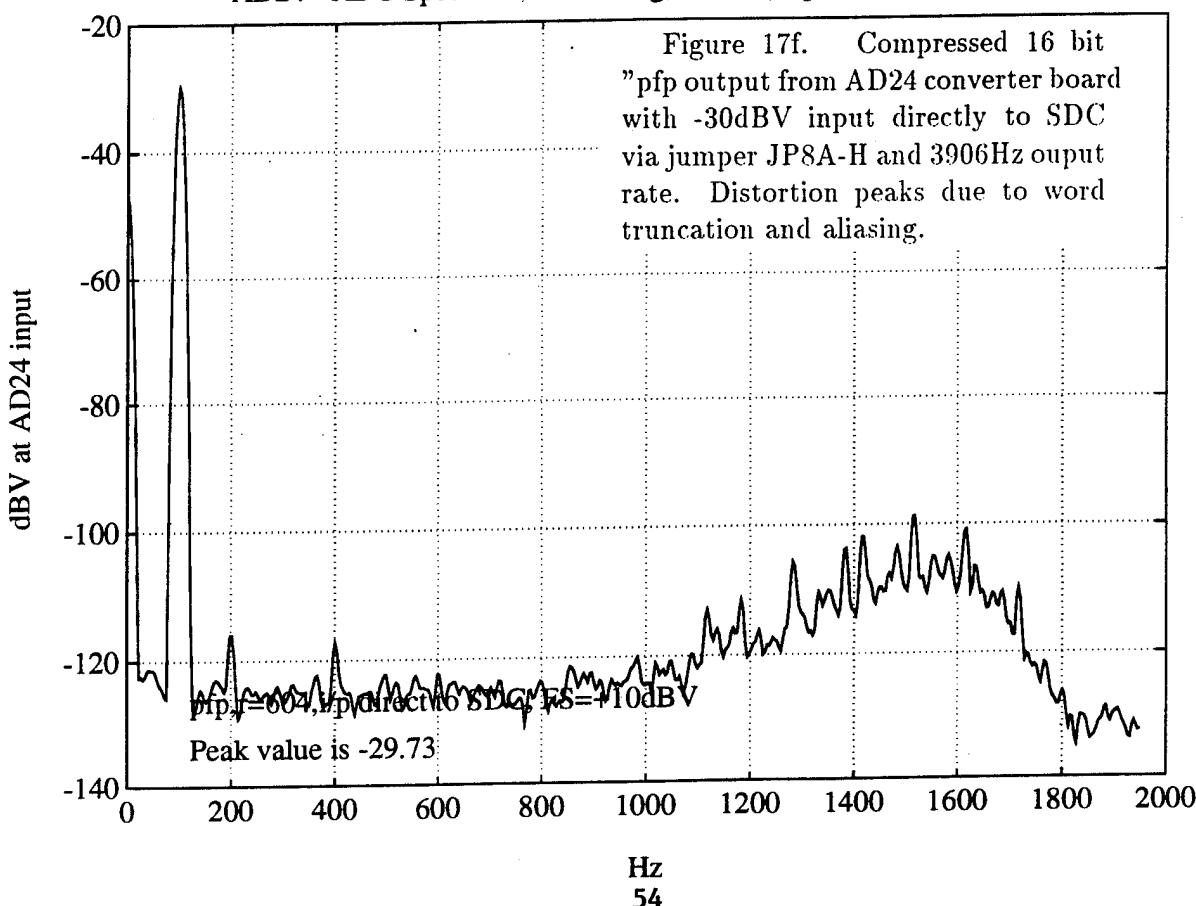
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



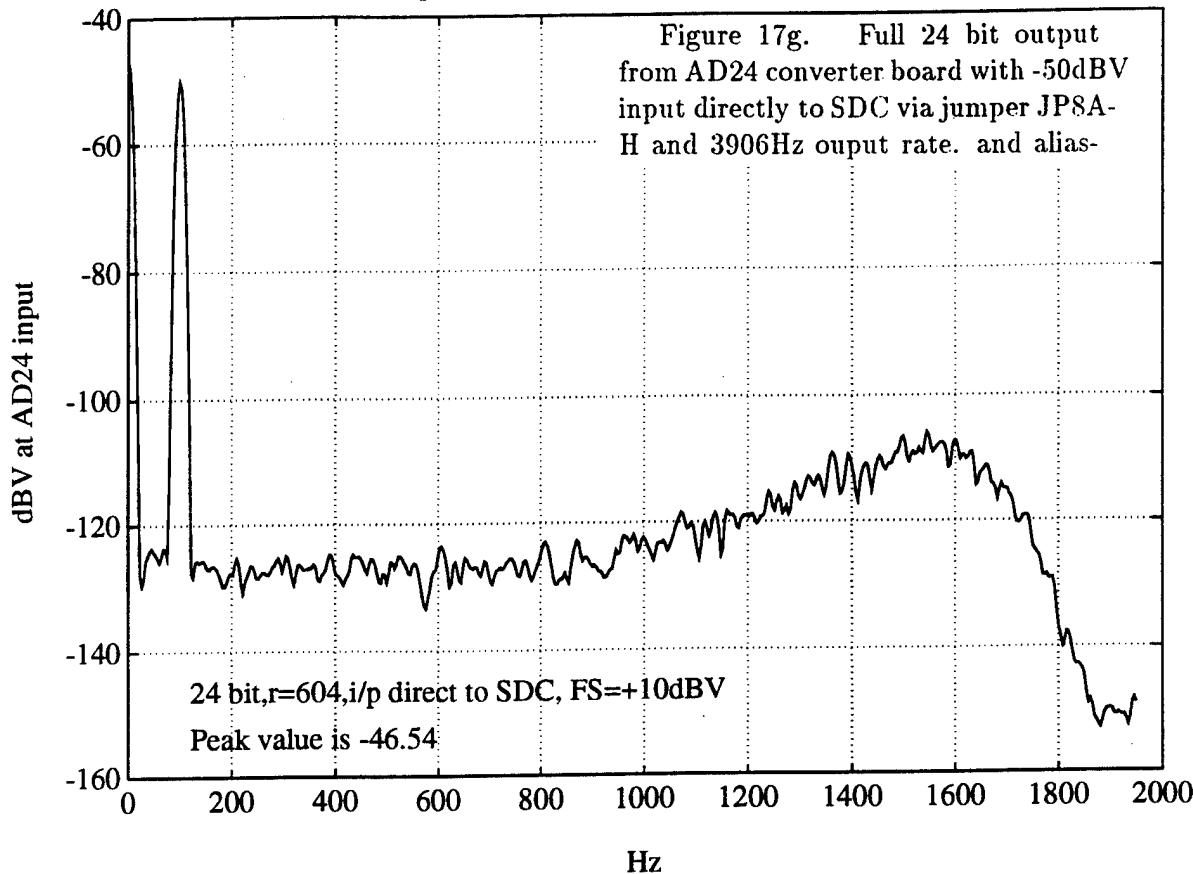
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



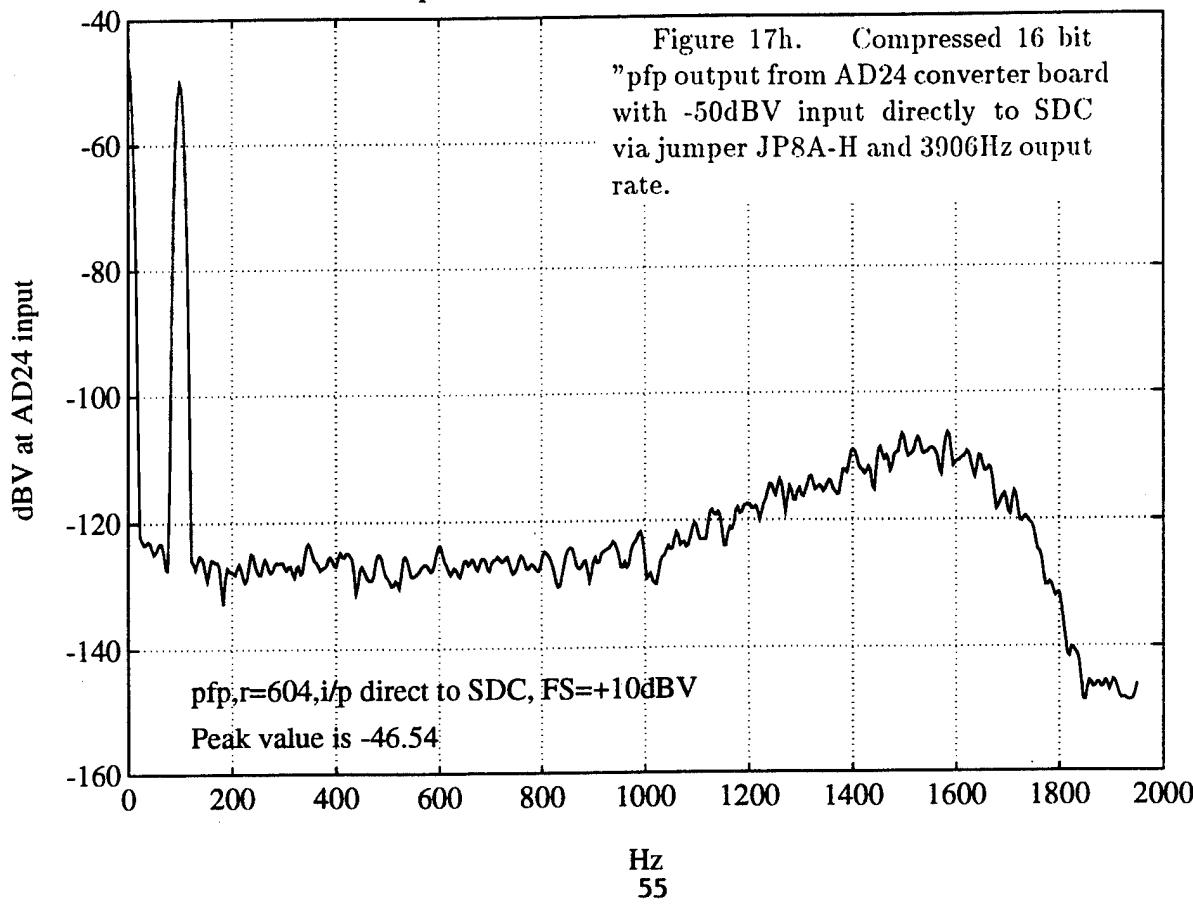
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



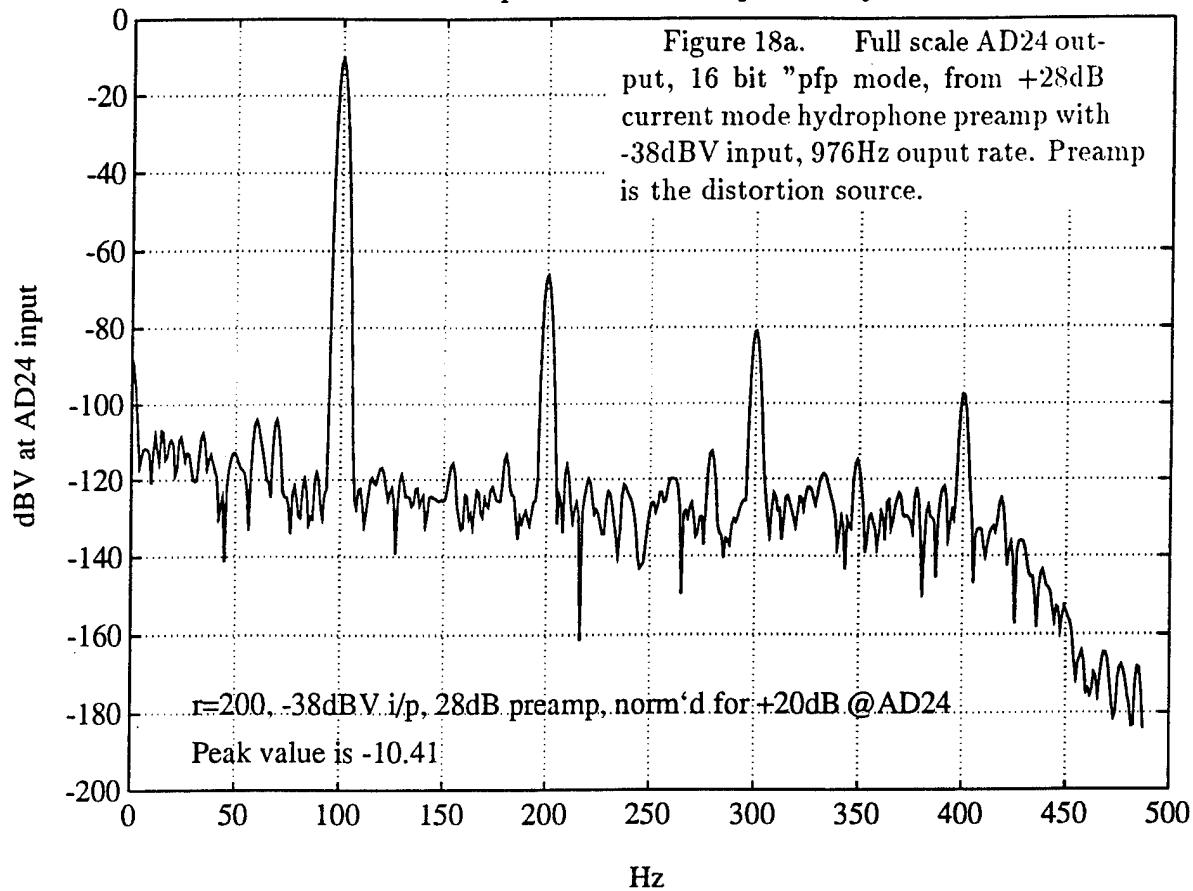
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



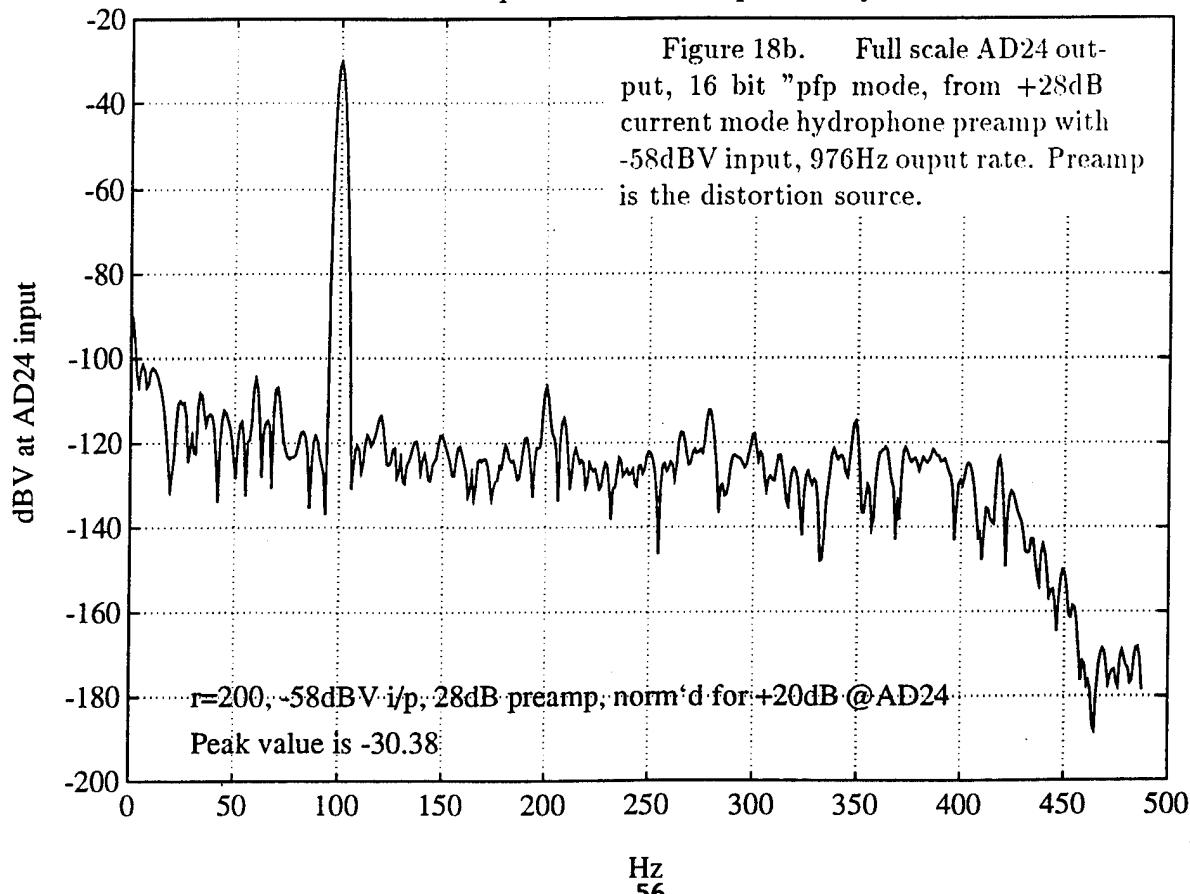
AD24 - ADC Spectrum, 8 - 1K avg, CH 1 ( 7 point Hody window )



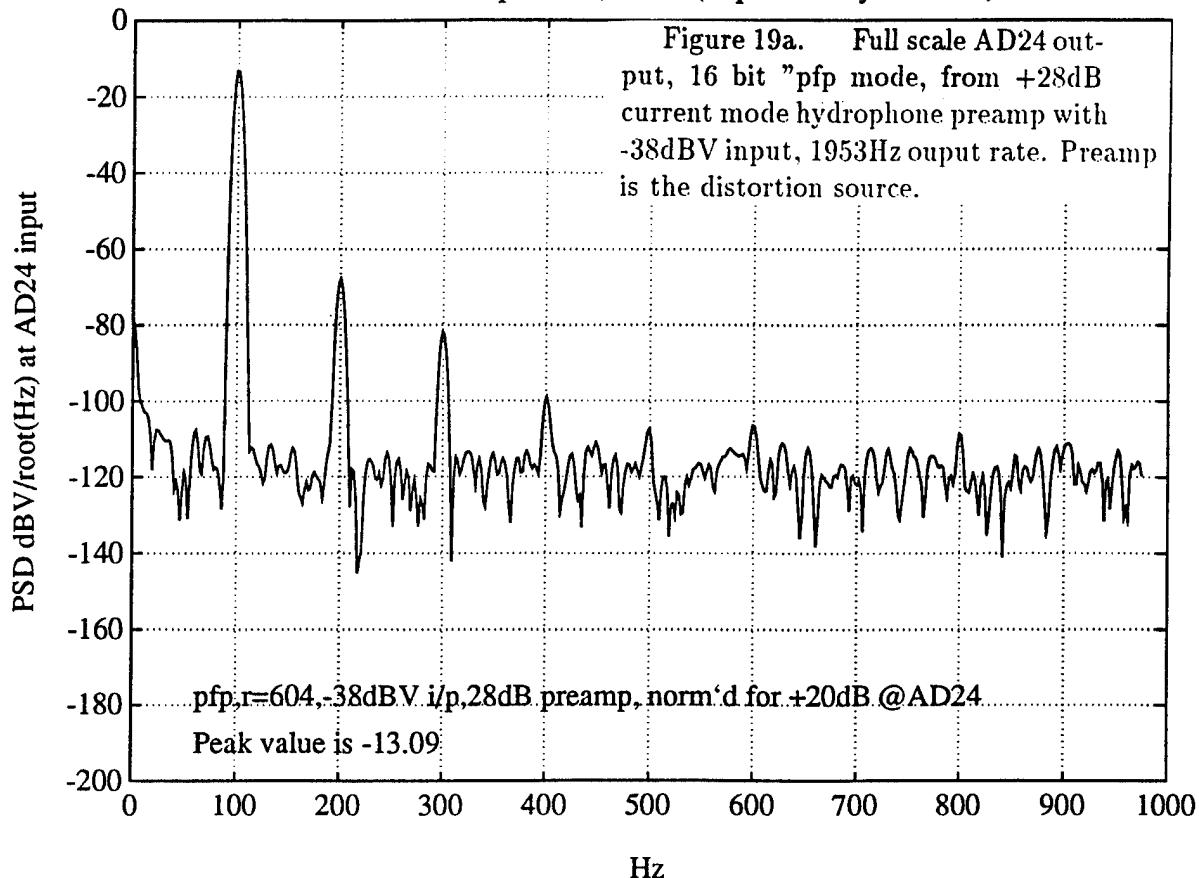
### AD24 - ADC Spectrum, CH 3 ( 7 point Hody window )



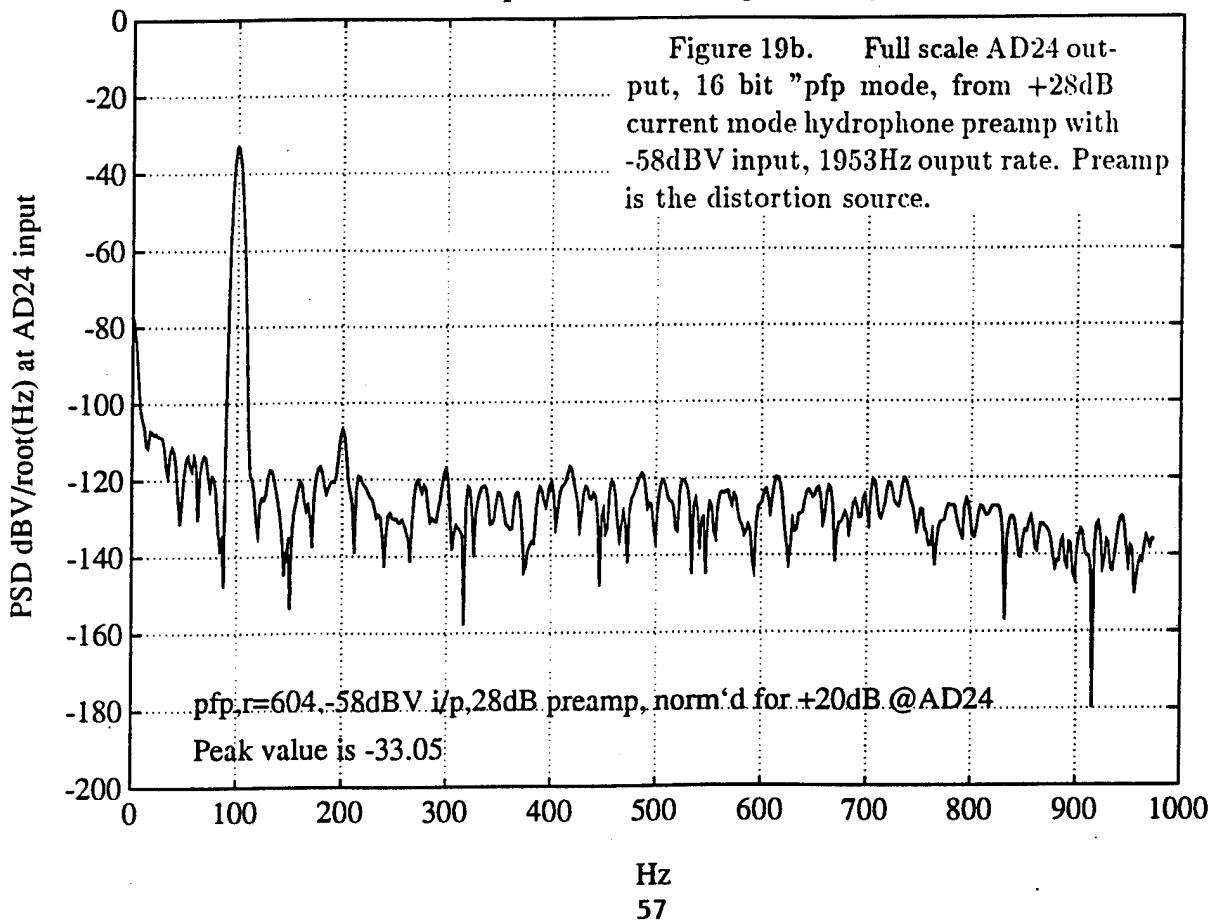
### AD24 - ADC Spectrum, CH 3 ( 7 point Hody window )



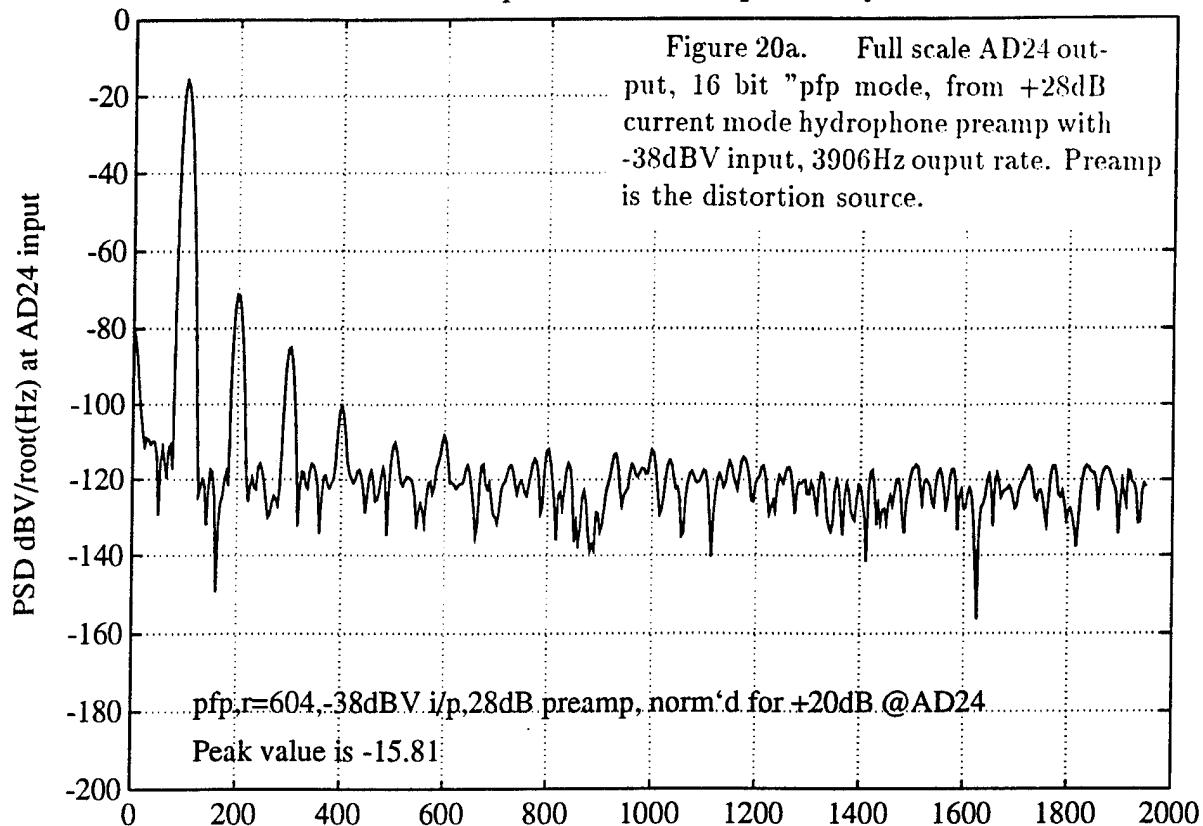
AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



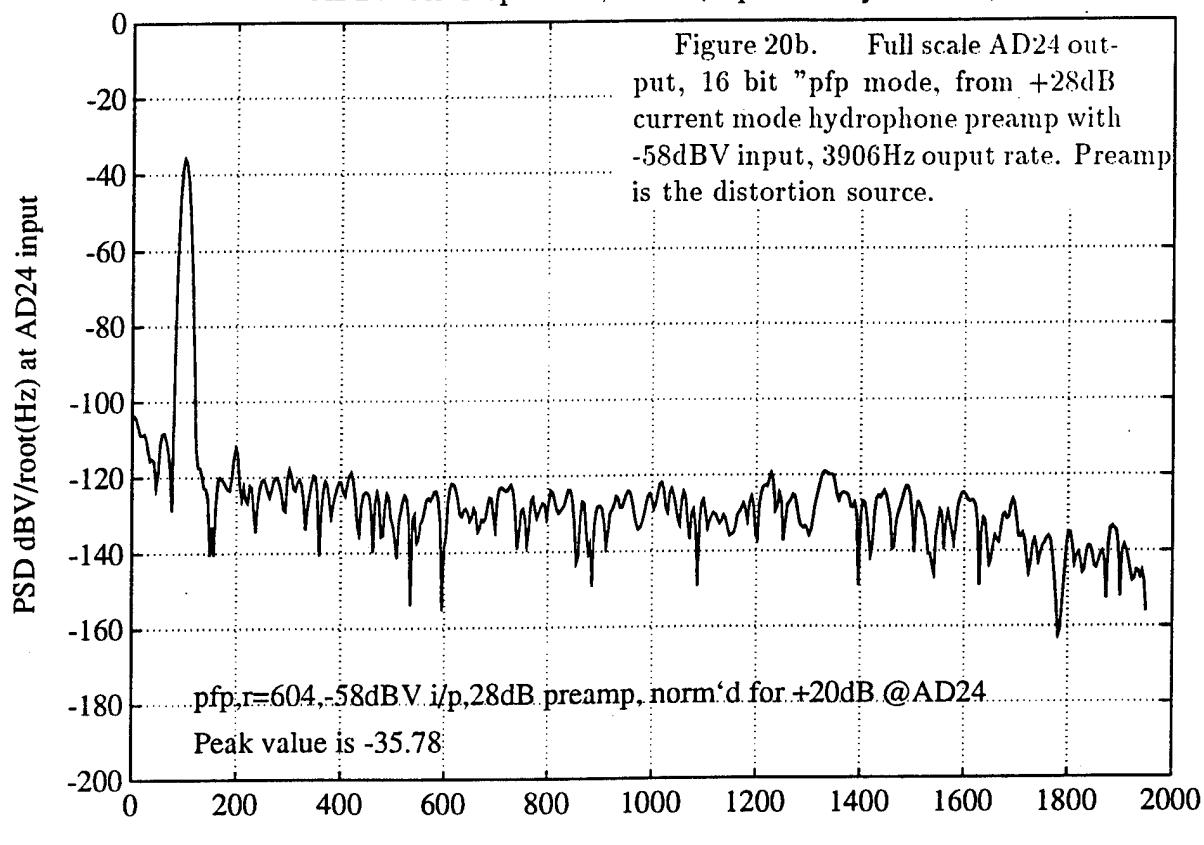
AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



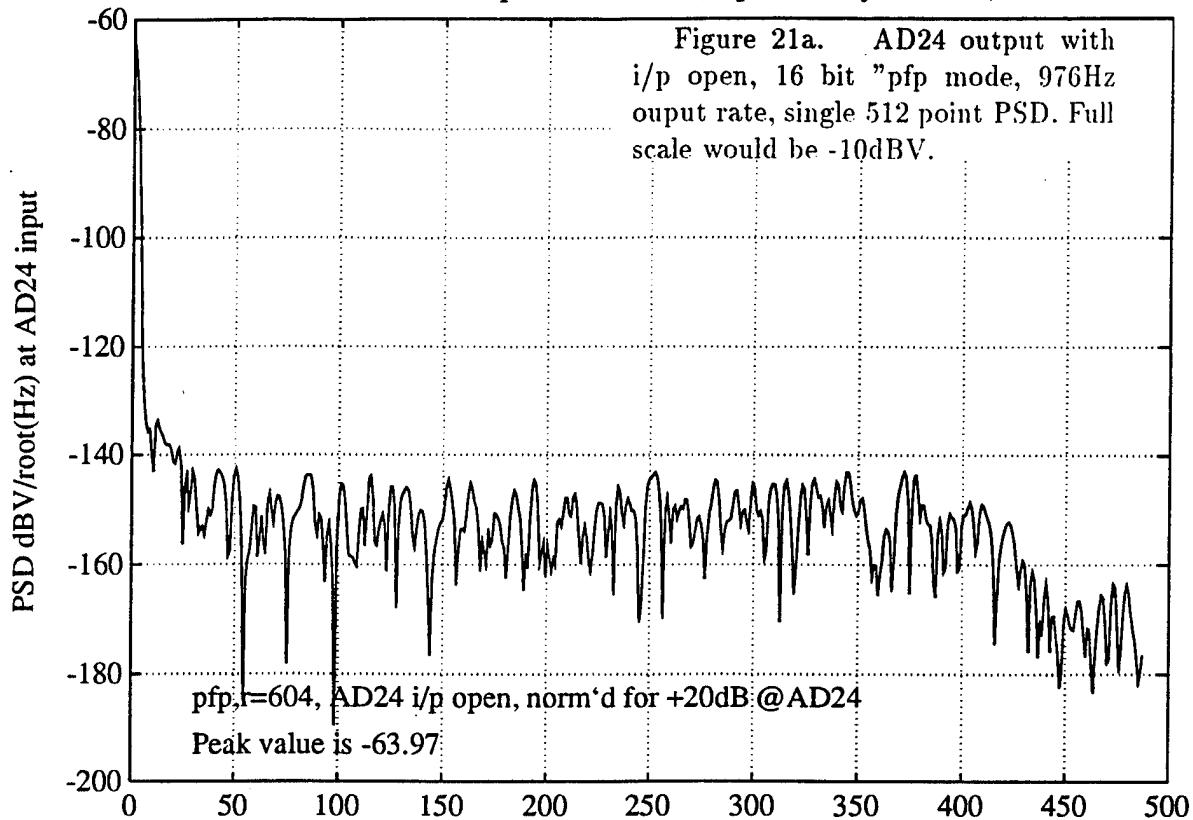
### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



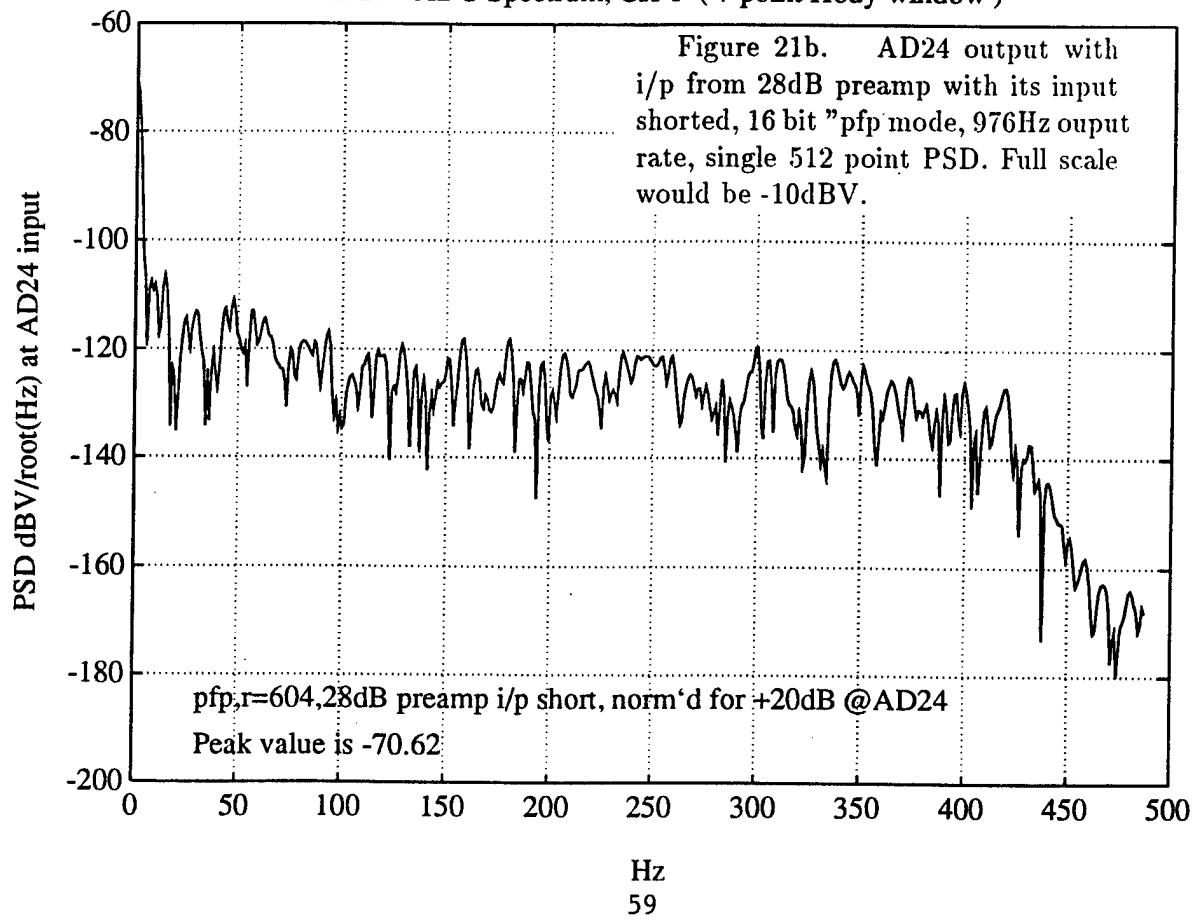
### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



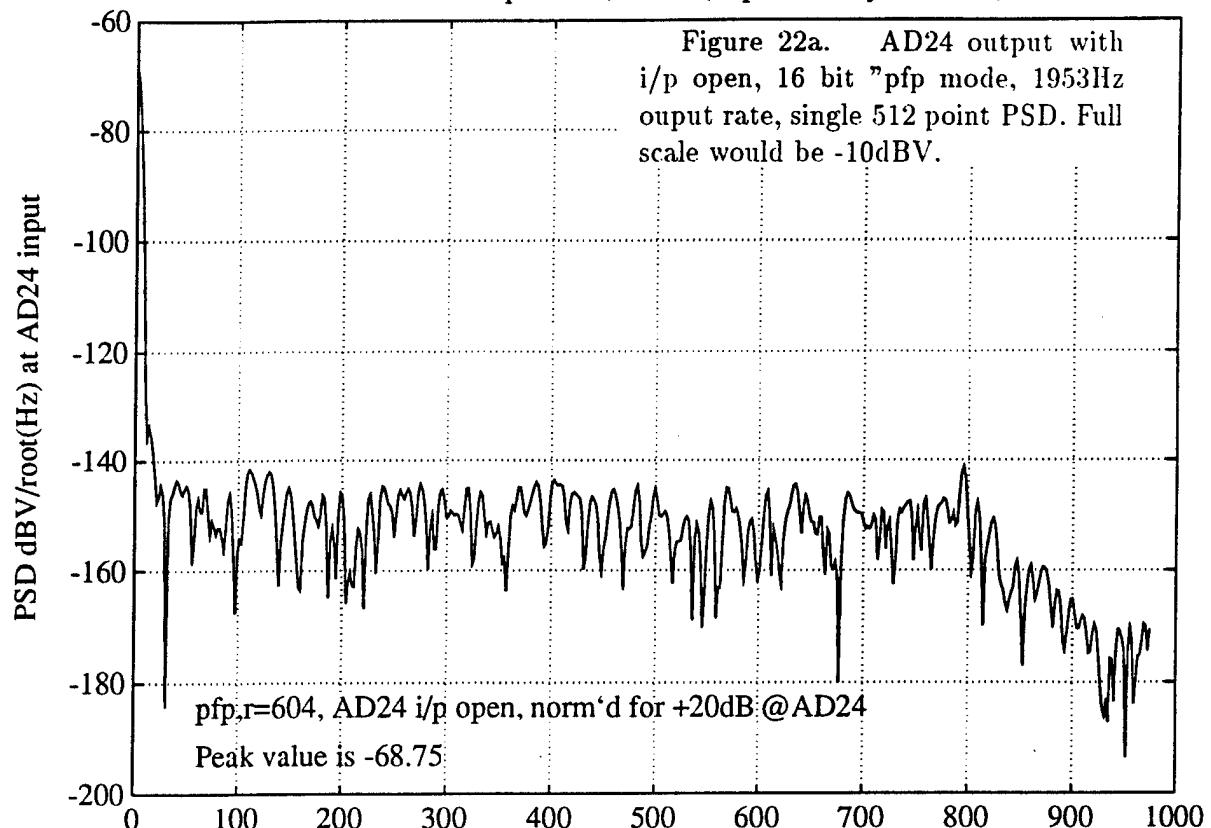
AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



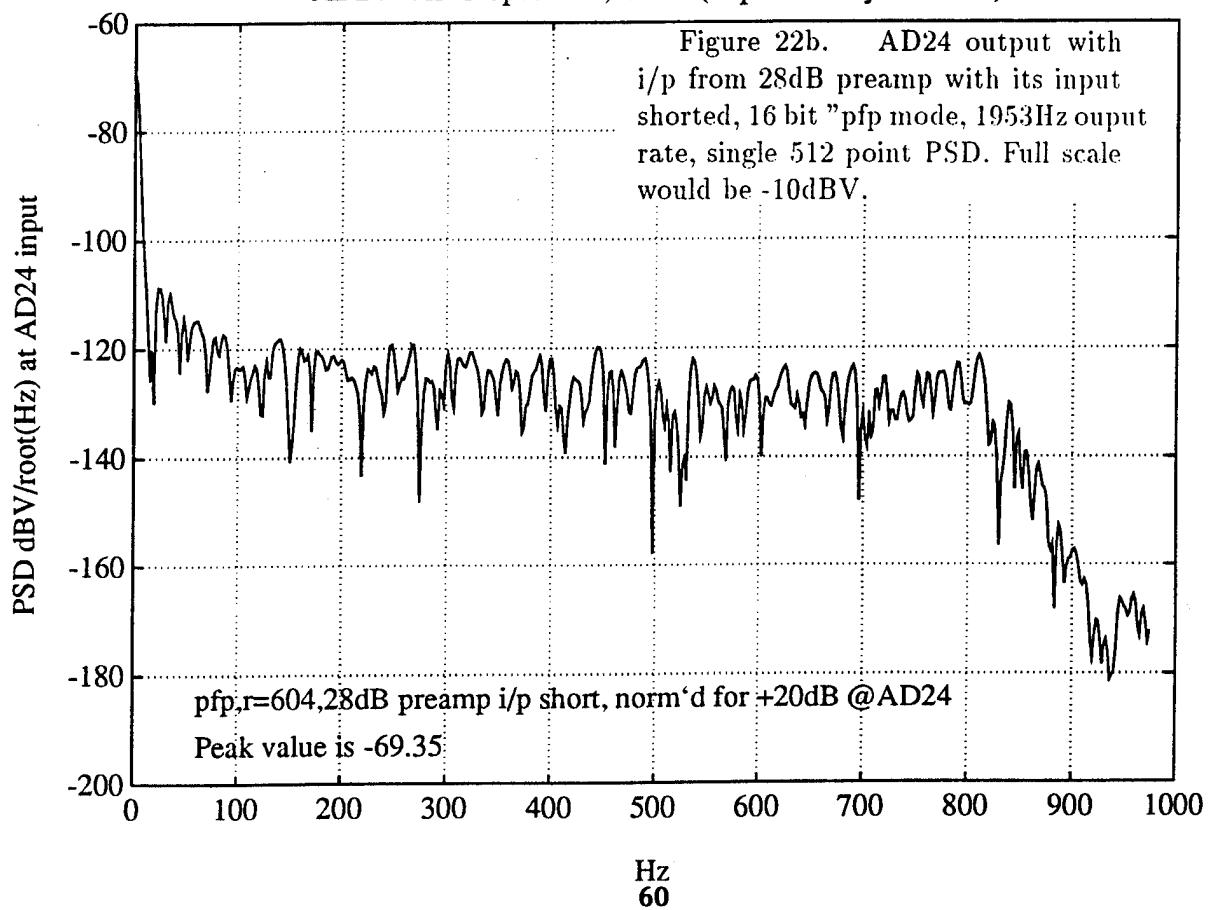
AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



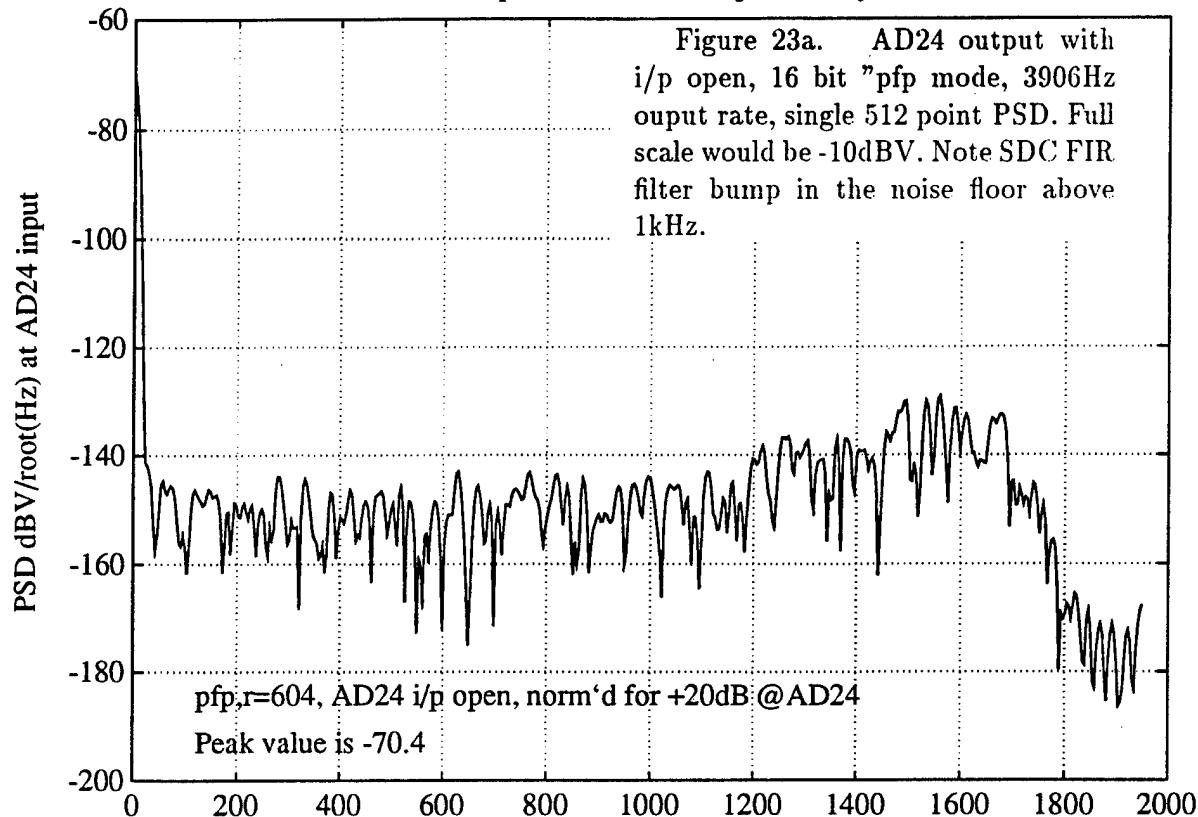
### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



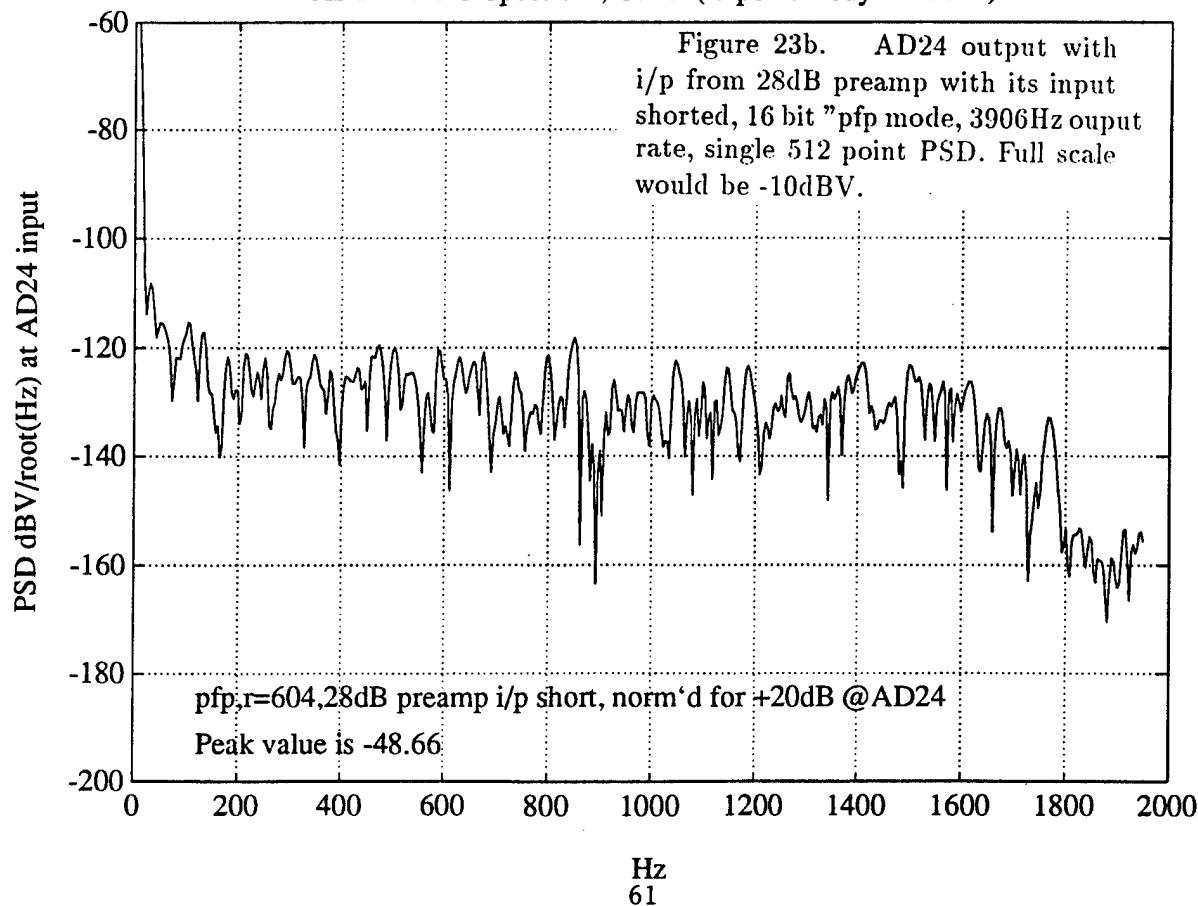
### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )

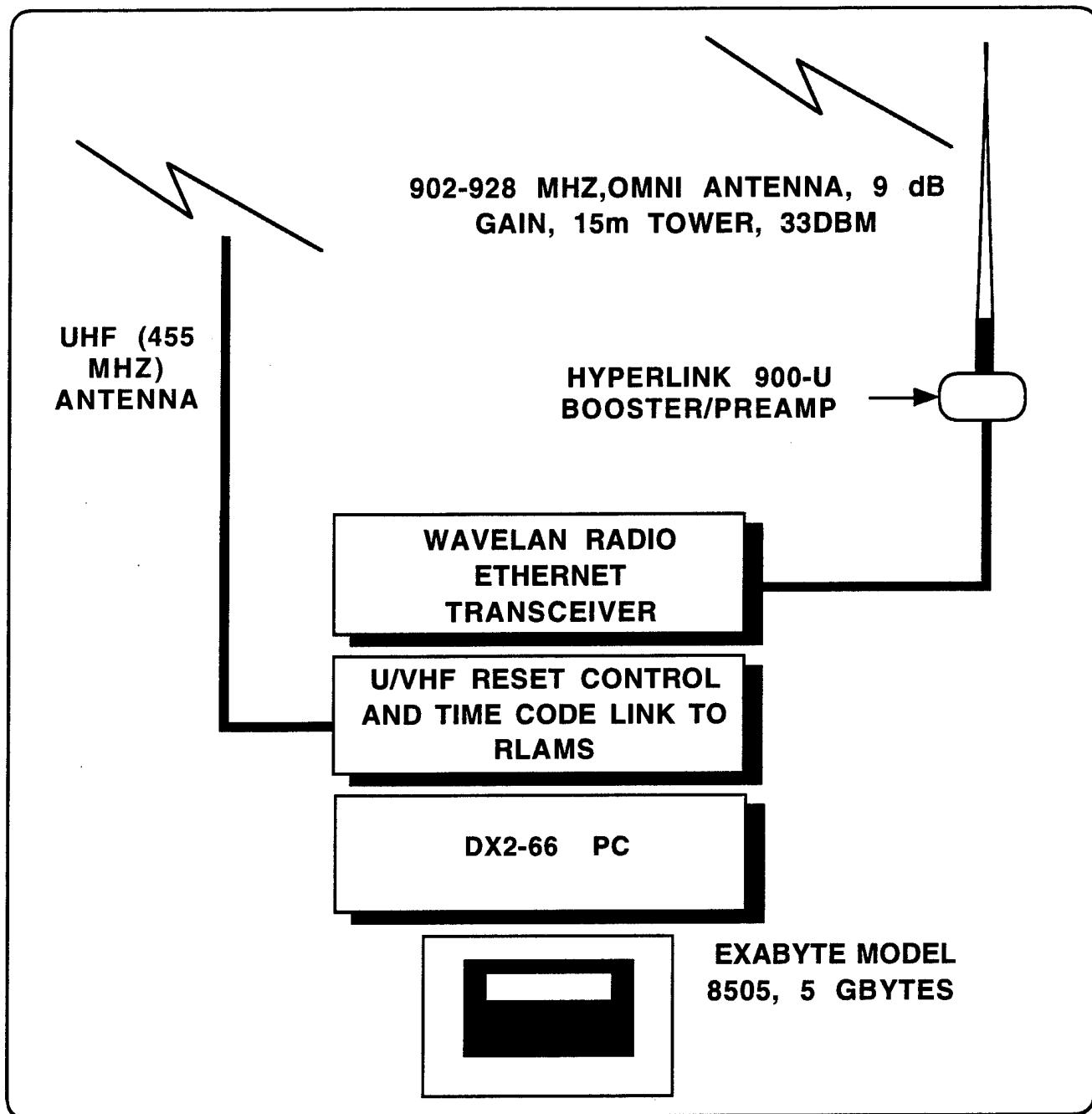


### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )



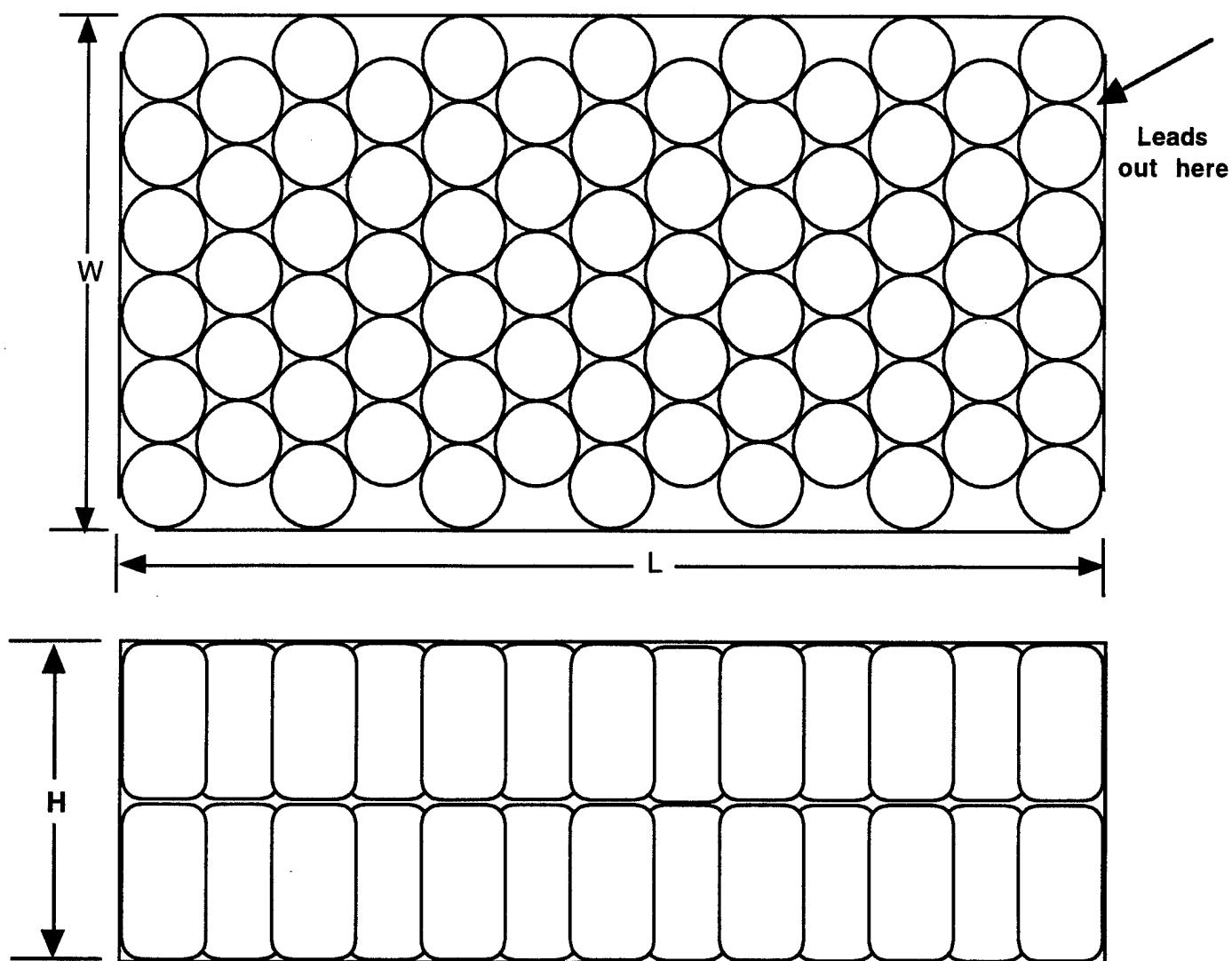
### AD24 - ADC Spectrum, CH 1 ( 7 point Hody window )





**Figure 24.** The receiving system for RLAM's monitors status of TCP connection, downloads command/control code, stores data from all RLAM systems to 8mm tape, performs simple checks on data quality and status of RLAM operation.

### Battery cell arrangement per layer of 2 layer deck



**Figure 25** RLAM battery. Each pack consists of 2 layers of 72 D-alkaline cells, arranged as shown in 7 rows of 6 plus 6 rows of 5 cells. The pack is connected as 6 parallel stacks of 24 cells each with an isolation diode (1N5822 Schottky or equivalent) resulting in a nominal 36 V. The finished dimensions are 14.9" L X 8.0" W X 4.9" H. The anticipated maximum continuous current from a single pack was about 1.4 amps at -10 degrees C. Leads are #16 or larger wire. Total estimated watt-hours per pack at 25C is about 2000; at 0C about 1500, assuming discharge to 18V. The manufacturer was Battery Assemblers Inc., Bohemia NY, 516 567 8855.

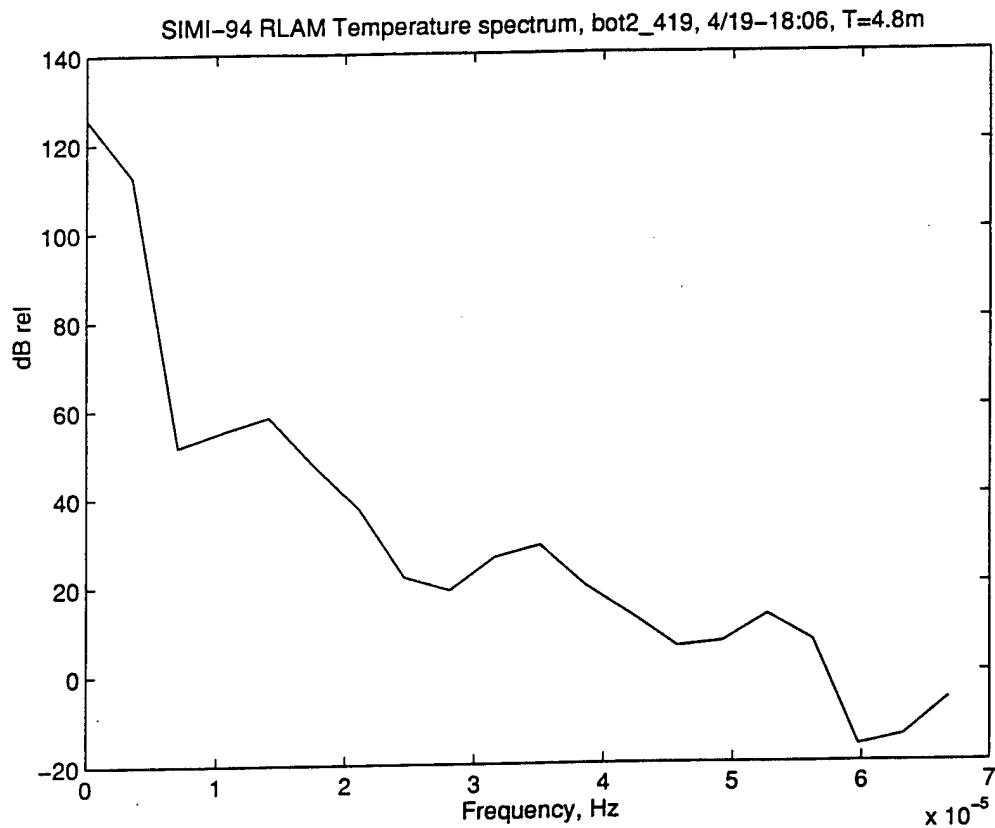
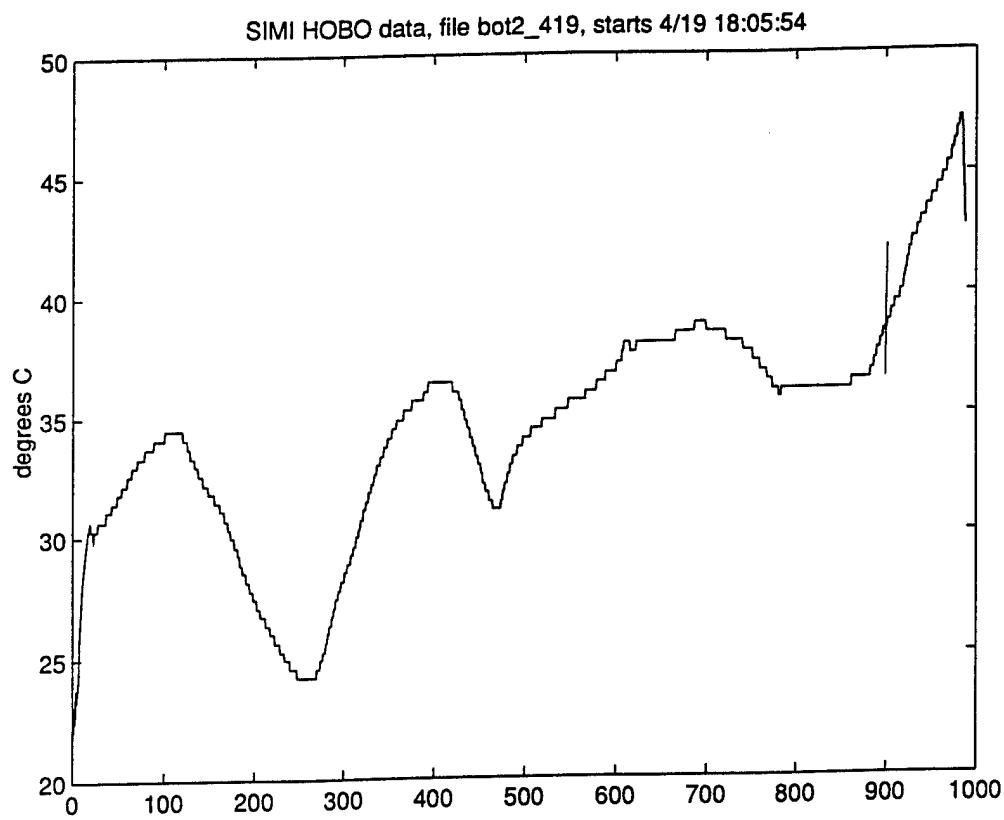


Figure 26a. RLAM-2, time series of internal temperature at 486 processor heat sink, near the bottom of the package, 4.8min sample period, SIMI-94 deployment. X-axis is sample count; vertical mark at about sample 900 indicates time that RLAM was brought into the hut. Figure 26b is the spectrum showing a peak near the daily frequency of .0000116 Hz.

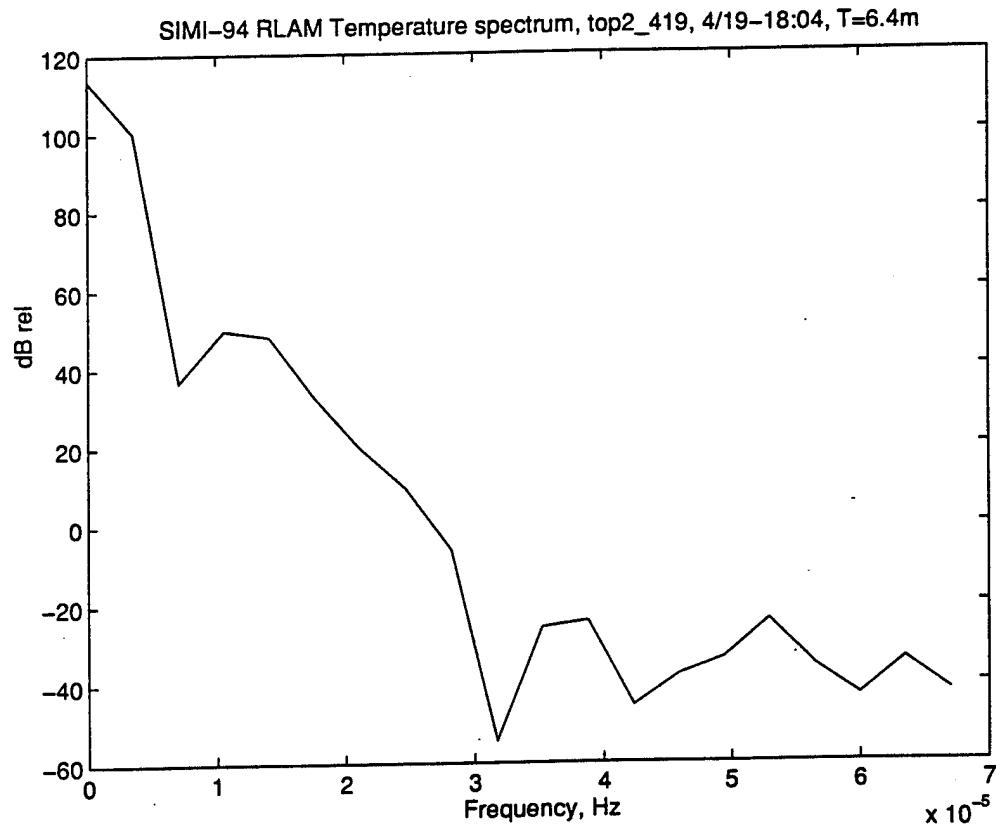
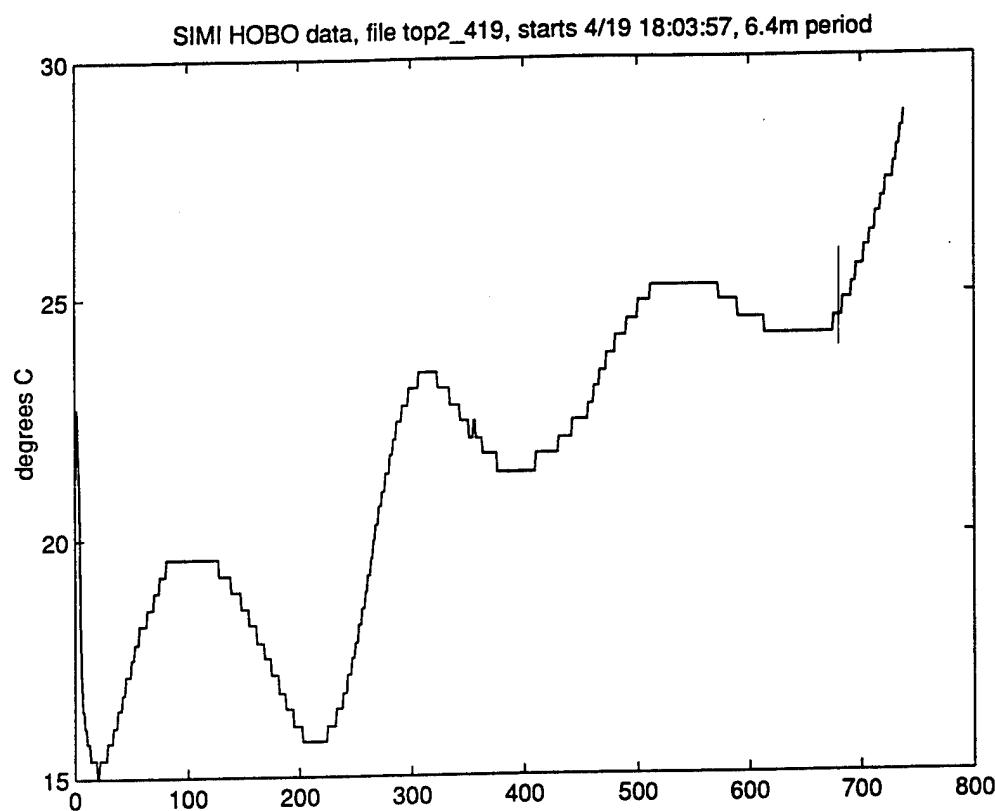


Figure 27a. RLAM-2, time series of internal temperature on top of one of the battterypacks, 6.4min sample period, SIMI-94 deployment. X-axis is sample count; vertical mark at about sample 900 indicates time that RLAM was brought into the hut. Figure 27b is the spectrum showing a peak near the daily frequency of .0000116 Hz.

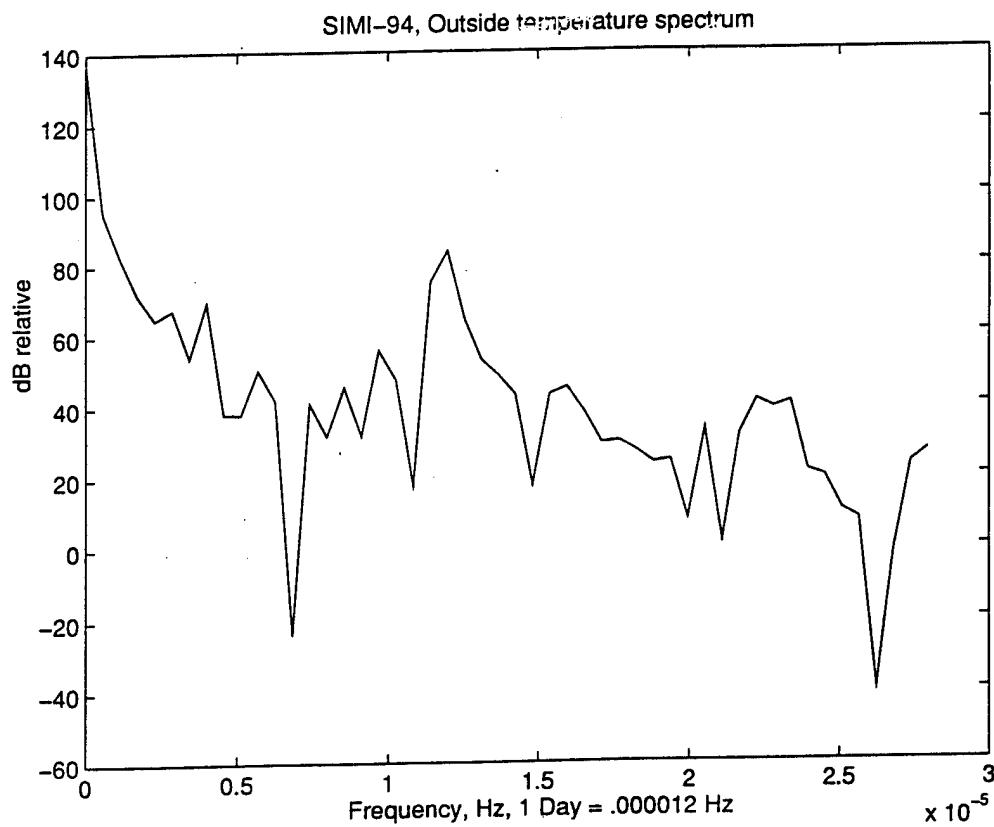
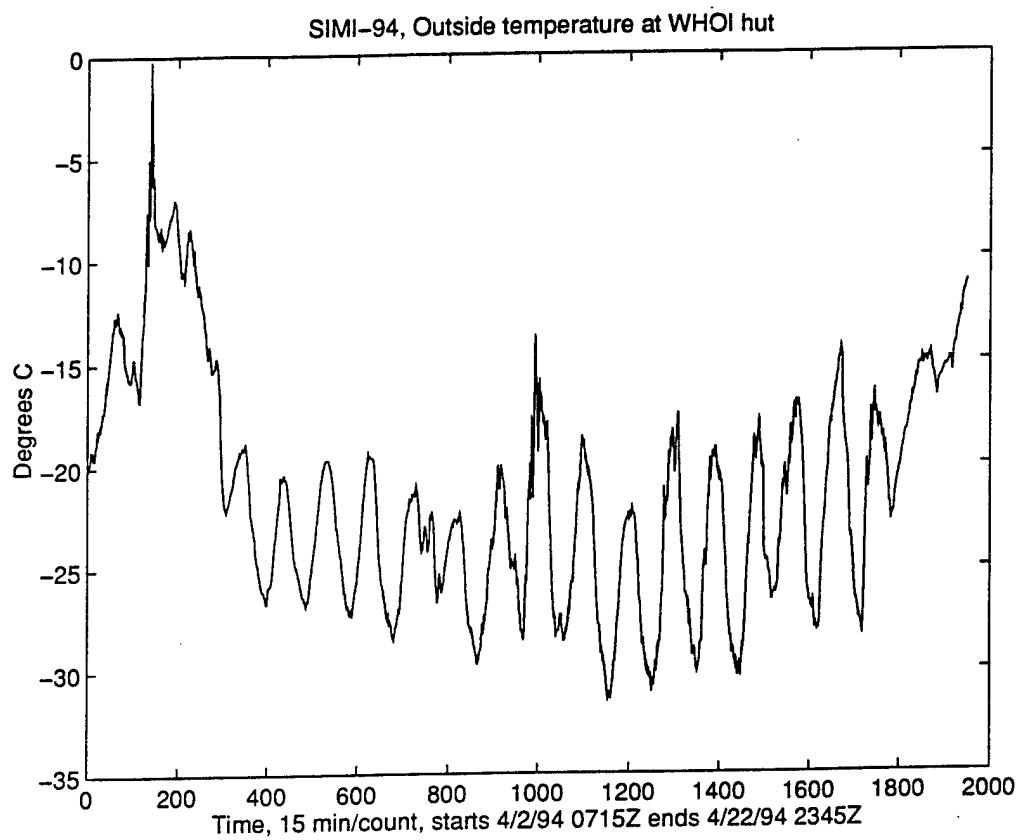
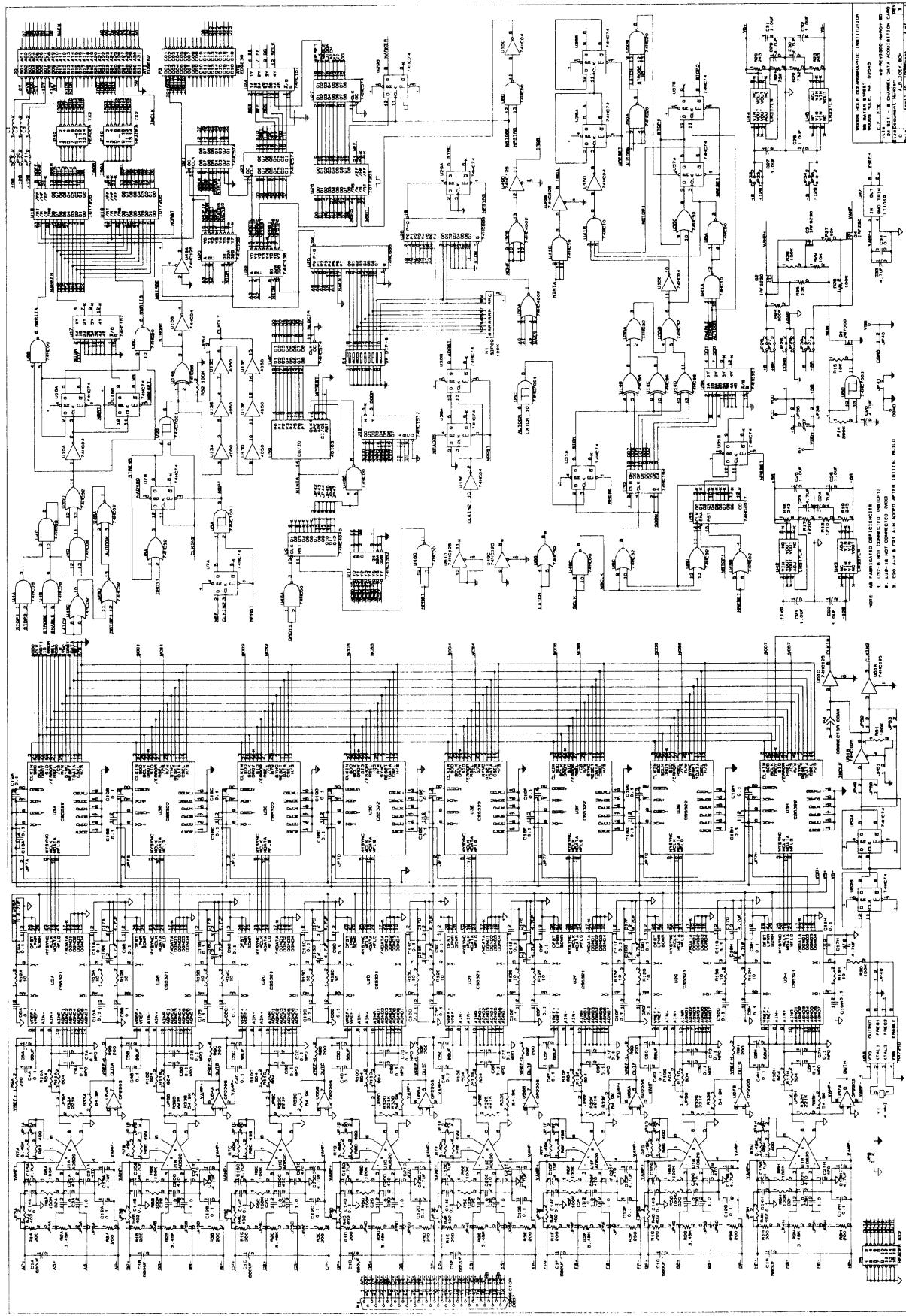


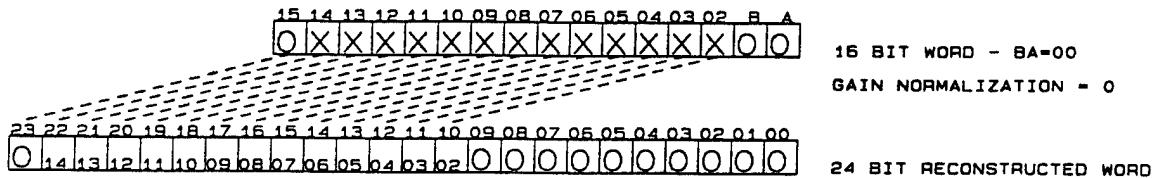
Figure 28a. Time series of outside ambient air temperature during SIMI-94, 2 - 22 April, 15 min sample period. Figure 28b is the spectrum showing a peak near the daily frequency of .0000116 Hz.

Figure 29. (continued) Schematic of the AD21 board.

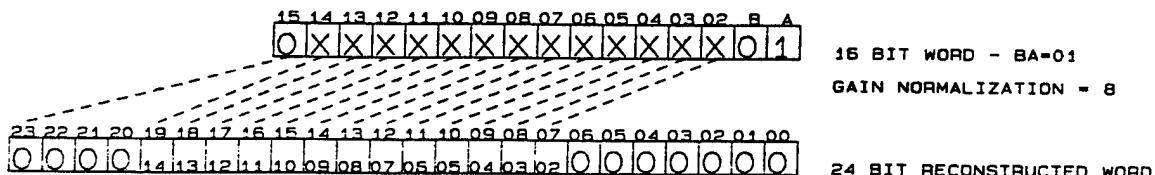


# 16 BIT PSEUDO FLOATING POINT WORD MAPPING TO 24 BITS

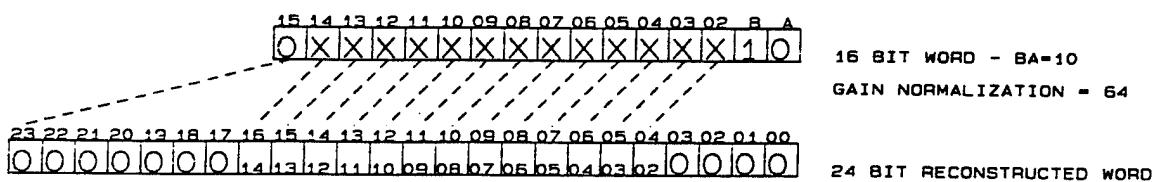
CASE #1 EXPONENT = 00



CASE #2 EXPONENT = 01



CASE #3 EXPONENT = 10



CASE #4 EXPONENT = 11

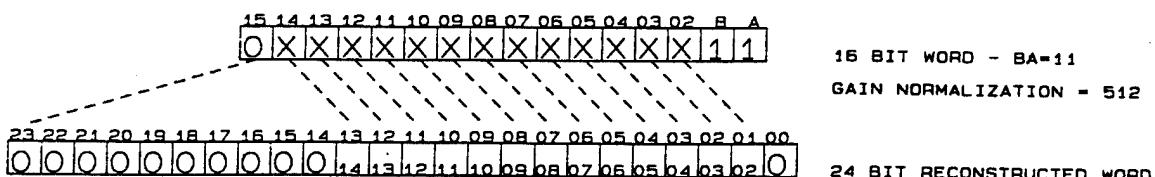


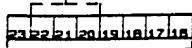
Figure 30. Mapping of 24 bit AD24 output to 16 bit pseudo floating point (pfp) mode 2.

24 BIT ADC WORD - TO - 16 BIT PFP WORD CONVERSION  
TESTING FOR ACTIVE BITS IN A 24 BIT WORD

CASE #1 EXPONENT = 00

SHIFT INTO TEST REGISTER THE MOST SIGNIFICANT 8 BITS FROM A/D (BITS 23 - 16)

ANY ACTIVE BITS ?



YES - SAVE AND STORE THESE 8 BITS & THEN SHIFT IN 8 MORE BITS (BITS 15 - 8)

SAVE & STORE



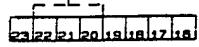
BA [ ] SAVE AND STORE THESE 8 BITS PLUS EXPONENT BA=00  
AS AN 8 BIT LSBYTE

SAVE & STORE

CASE #2 EXPONENT = 01

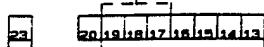
SHIFT INTO TEST REGISTER THE MOST SIGNIFICANT 8 BITS FROM A/D (BITS 23 - 16)

ANY ACTIVE BITS ?



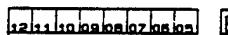
NO - SHIFT IN 3 MORE BITS (BITS 15 - 13) - INCREMENT EXPONENT COUNT BA

ANY ACTIVE BITS ?



YES - SAVE SIGN PLUS THESE 7 BITS & THEN SHIFT IN 8 MORE BITS (BITS 12 - 05)

SAVE & STORE



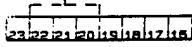
BA [ ] SAVE AND STORE THESE 8 BITS PLUS EXPONENT BA=01  
AS AN 8 BIT LSBYTE

SAVE & STORE

CASE #3 EXPONENT = 10

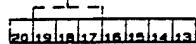
SHIFT INTO TEST REGISTER THE MOST SIGNIFICANT 8 BITS FROM A/D (BITS 23 - 16)

ANY ACTIVE BITS ?



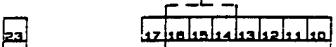
NO - SHIFT IN 3 MORE BITS (BITS 15 - 13) - INCREMENT EXPONENT COUNT BA

ANY ACTIVE BITS ?



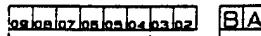
NO - SHIFT IN 3 MORE BITS (BITS 12 - 10) - INCREMENT EXPONENT COUNT BA

ANY ACTIVE BITS ?



YES - SAVE SIGN PLUS THESE 7 BITS & THEN SHIFT IN 8 MORE BITS (BITS 09 - 02)

SAVE & STORE

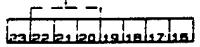


BA [ ] SAVE AND STORE THESE 8 BITS PLUS EXPONENT  
BA=10 AS AN 8 BIT LSBYTE

SAVE & STORE

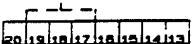
CASE #4 EXPONENT = 11

ANY ACTIVE BITS ?



NO - SHIFT IN 3 MORE BITS (BITS 15 - 13) - INCREMENT EXPONENT COUNT BA

ANY ACTIVE BITS ?



NO - SHIFT IN 3 MORE BITS (BITS 12 - 10) - INCREMENT EXPONENT COUNT BA

ANY ACTIVE BITS ?



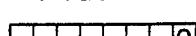
NO - SHIFT IN 3 MORE BITS (BITS 09 - 07) - INCREMENT EXPONENT COUNT BA

NO TEST



SAVE SIGN PLUS THESE 7 BITS & THEN SHIFT IN 8 MORE BITS (BITS 06 - 00)  
(THE LAST SHIFT DOES NOT BRING IN DATA BUT A TRAILING 0 INSTEAD)

SAVE & STORE



BA [ ] SAVE AND STORE THESE 8 BITS PLUS  
EXPONENT BA=11 AS AN 8 BIT LSBYTE

SAVE & STORE

NOTE - FOR NEGATIVE (2'S COMPLEMENT) NUMBERS, ACTIVE BIT TEST IS FOR 0'S INSTEAD OF 1'S  
A 1 IN THE SIGN BIT AUTOMATICALLY CHANGES THE CIRCUITRY TO TEST FOR 0'S

Figure 31. AD24 algorithm for conversion of 24 bit AD24 output to 16 bit pseudo floating point (pfp) mode 2.

**FILTER CHARACTERISTICS** (TA = T<sub>min</sub> to T<sub>max</sub>; VD+ = 5V; GND = 0V; CLKIN = 1.024 MHz; transfer function shown in Figure 2; unless otherwise specified.)

Output Word Rate f <sub>0</sub> (Hz)	Passband f <sub>1</sub> (Hz)	Passband Flatness R <sub>FB</sub> (dB)	-3dB Freq. f <sub>2</sub> (Hz)	Stopband f <sub>3</sub> (Hz) (Note 13)	Group Delay (ms)
4000	1500	0.2	1652.5	2000	7.25
2000	750	0.04	824.3	1000	14.5
1000	375	0.08	411.9	500	29
500	187.5	0.1	205.9	250	58
250	93.8	0.1	102.9	125	116
125	46.9	0.1	51.5	62.5	232
62.5	23.4	0.1	25.7	31.25	464

Note: 13. G<sub>SB</sub> = -130 dB for all Output Word Rates.

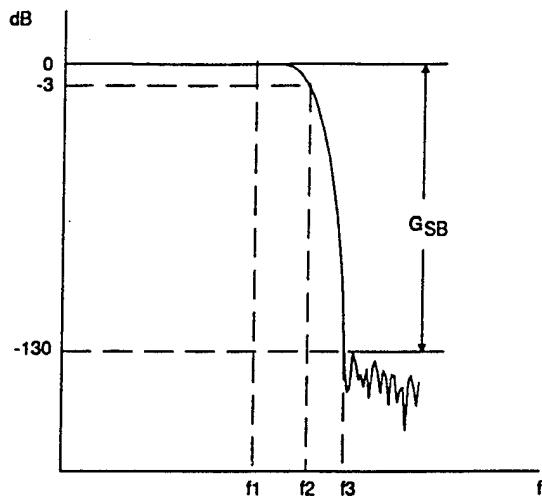


Figure 1. CS5322 Filter Response

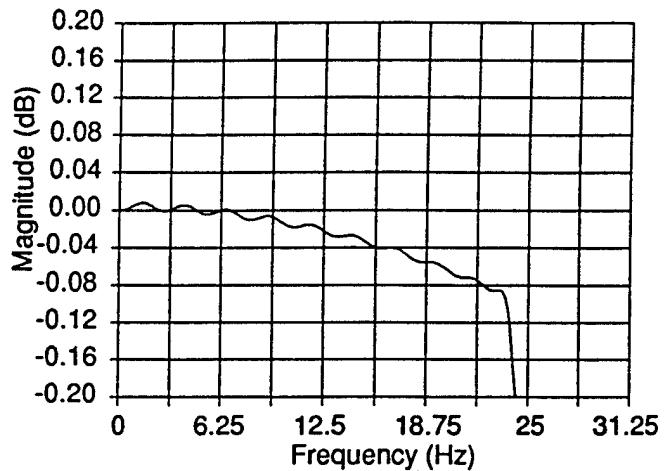


Figure 2. CS5322 Digital Filter Passband Ripple  
f<sub>0</sub> = 62.5 Hz

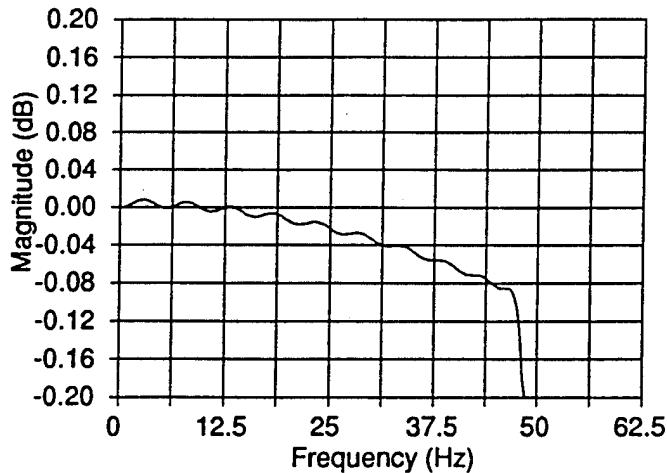


Figure 3. CS5322 Digital Filter Passband Ripple  
f<sub>0</sub> = 125Hz

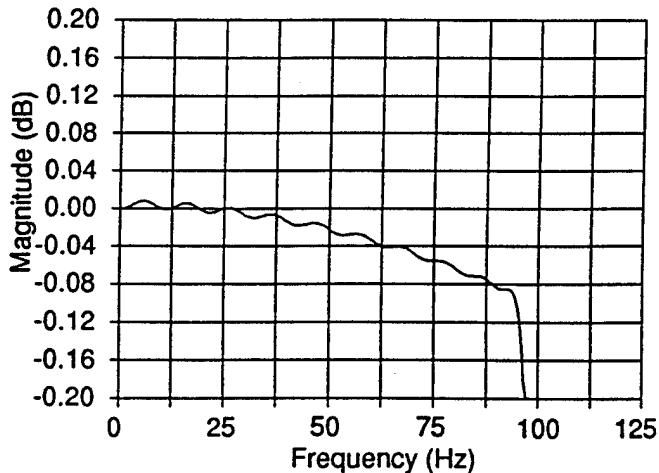


Figure 4. CS5322 Digital Filter Passband Ripple  
f<sub>0</sub> = 250 Hz

Figure 32a. Crystal Semiconductor CS5322 FIR filter characteristics.

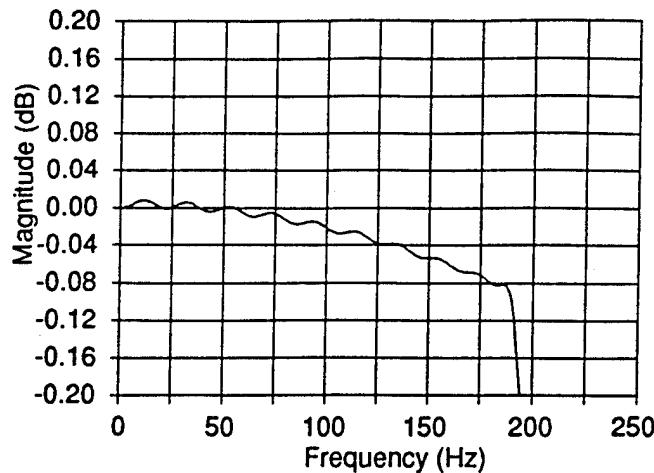


Figure 5. CS5322 Digital Filter Passband Ripple  
 $f_0 = 500$  Hz

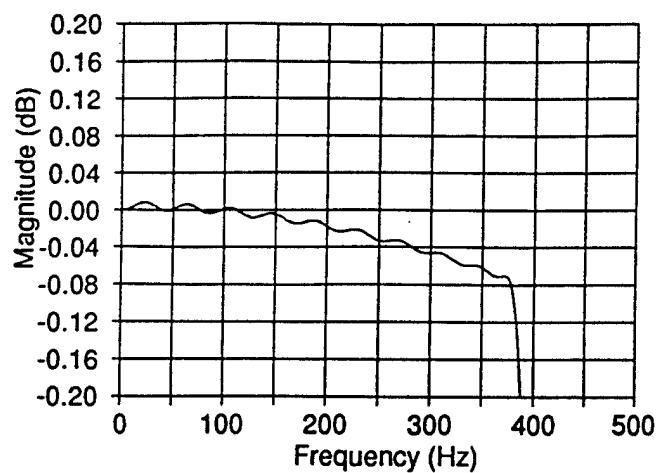


Figure 6. CS5322 Digital Filter Passband Ripple  
 $f_0 = 1000$  Hz

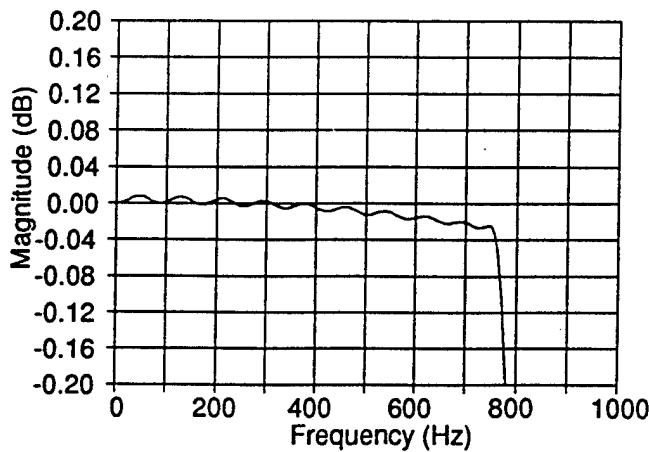


Figure 7. CS5322 Digital Filter Passband Ripple  
 $f_0 = 2000$  Hz

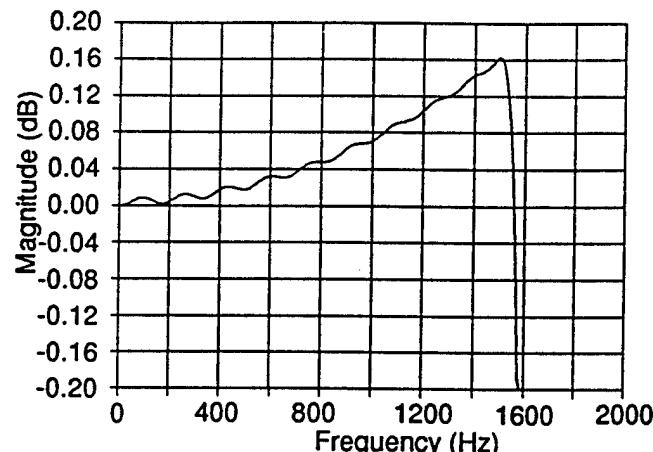


Figure 8. CS5322 Digital Filter Passband Ripple  
 $f_0 = 4000$  Hz

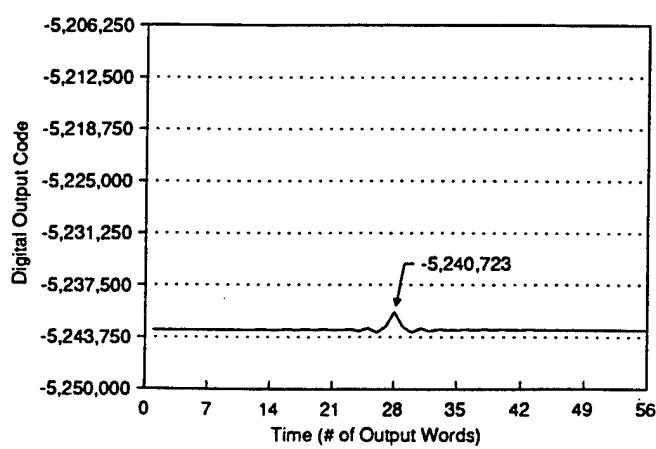


Figure 9. CS5322 Impulse Response,  $f_0 = 62.5$  Hz

Figure 32b. Crystal Semiconductor CS5322 FIR filter characteristics, showing constant group delay of 28 samples.

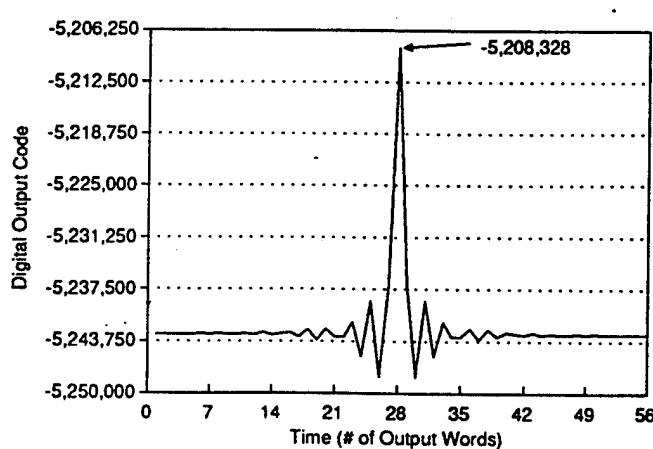
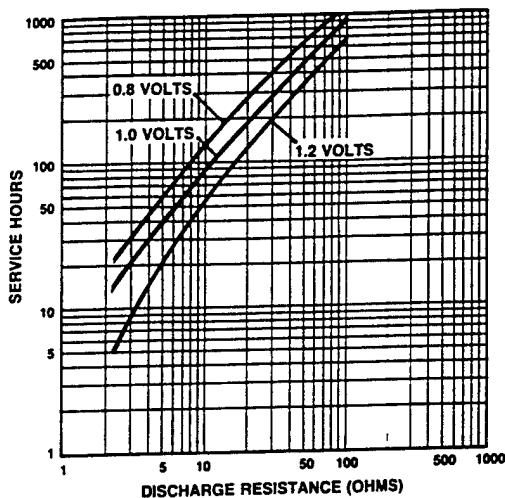
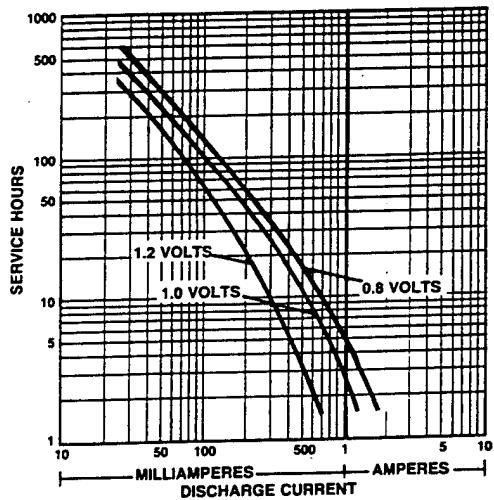


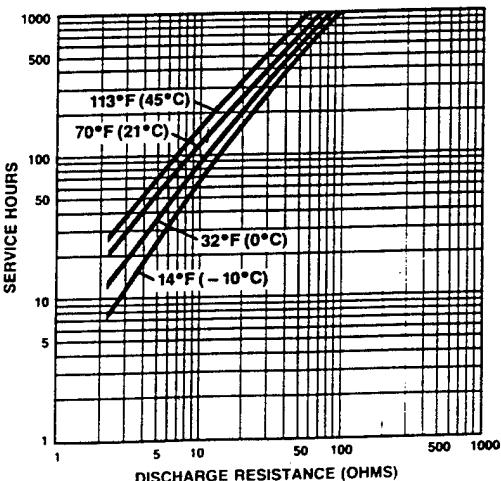
Figure 10. CS5322 Impulse Response,  $f_0 = 1000$  Hz



**TYPICAL CONTINUOUS PERFORMANCE CHARACTERISTICS WITH CONSTANT RESISTANCE AT 70°F (21°C)**



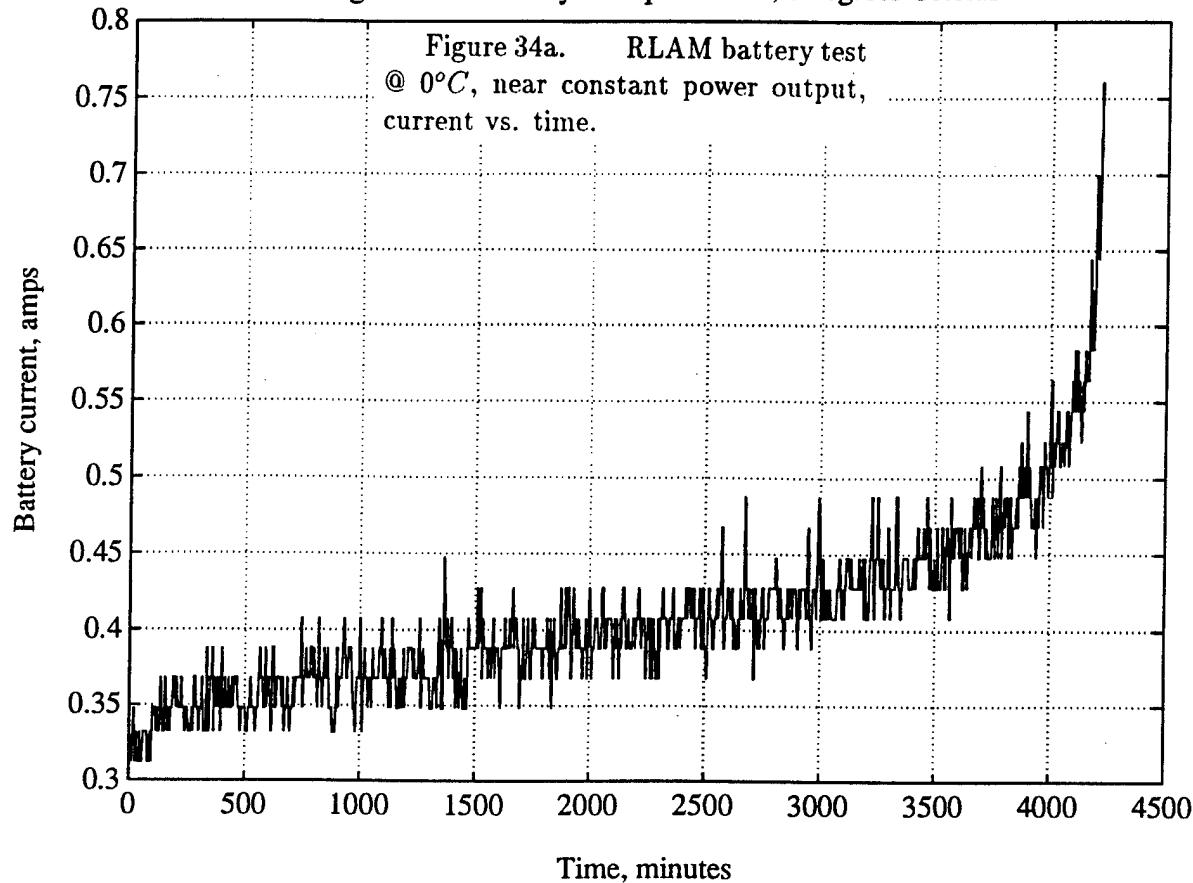
**TYPICAL CONTINUOUS PERFORMANCE CHARACTERISTICS WITH CONSTANT CURRENT AT 70°F (21°C)**



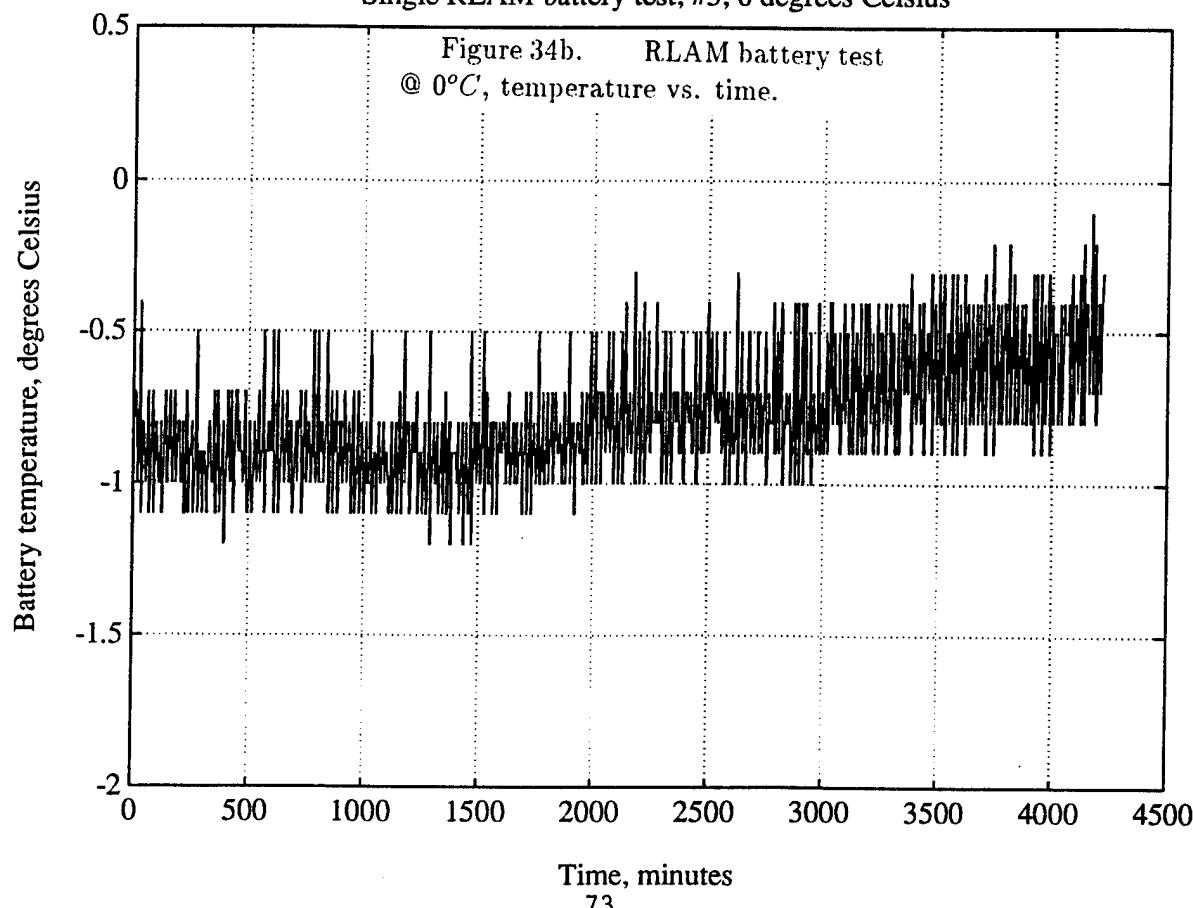
**TYPICAL CONTINUOUS PERFORMANCE CHARACTERISTICS WITH CONSTANT RESISTANCE AT VARIOUS TEMPERATURES TO 0.8 VOLTS**

Figure 33. Duracell data showing discharge characteristics of "D" size alkaline 1.5V cell.

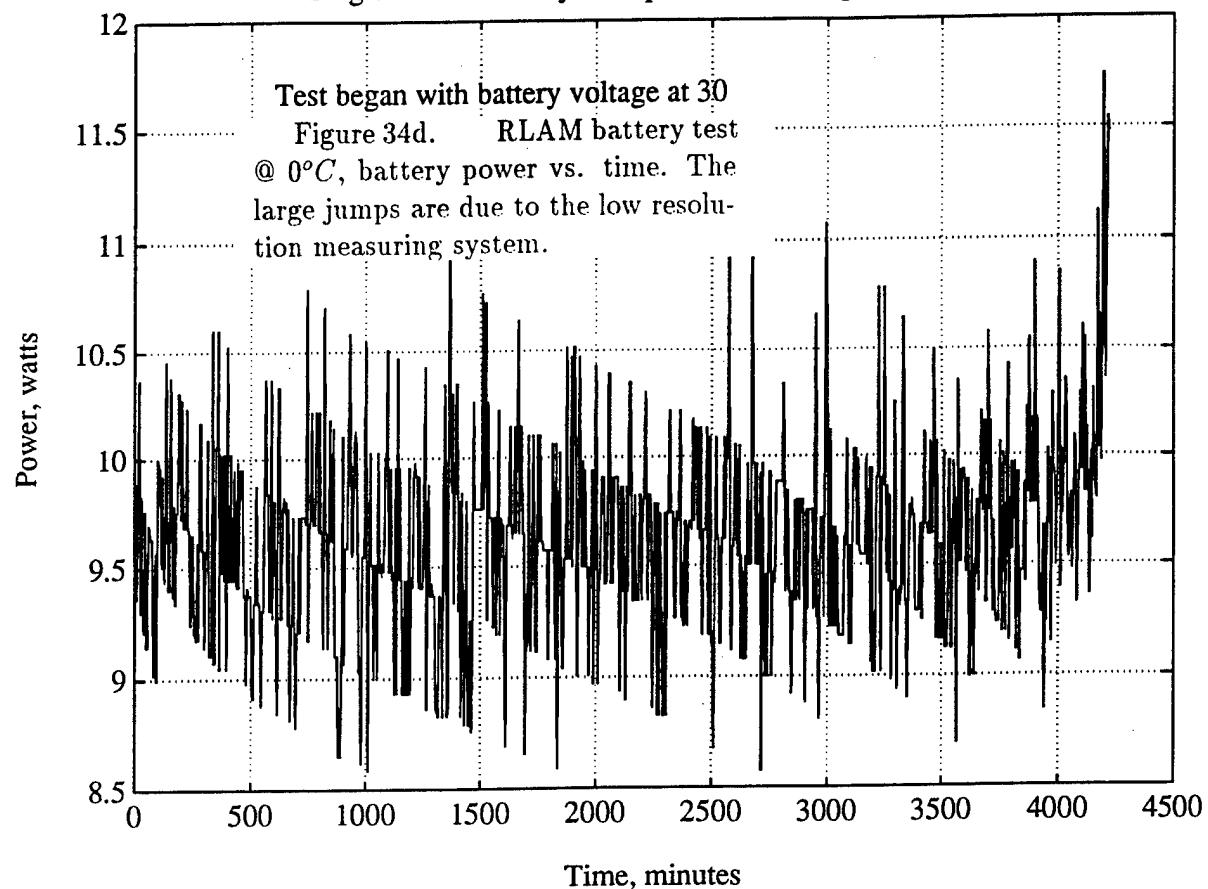
Single RLAM battery after partial use, 0 degrees Celcius



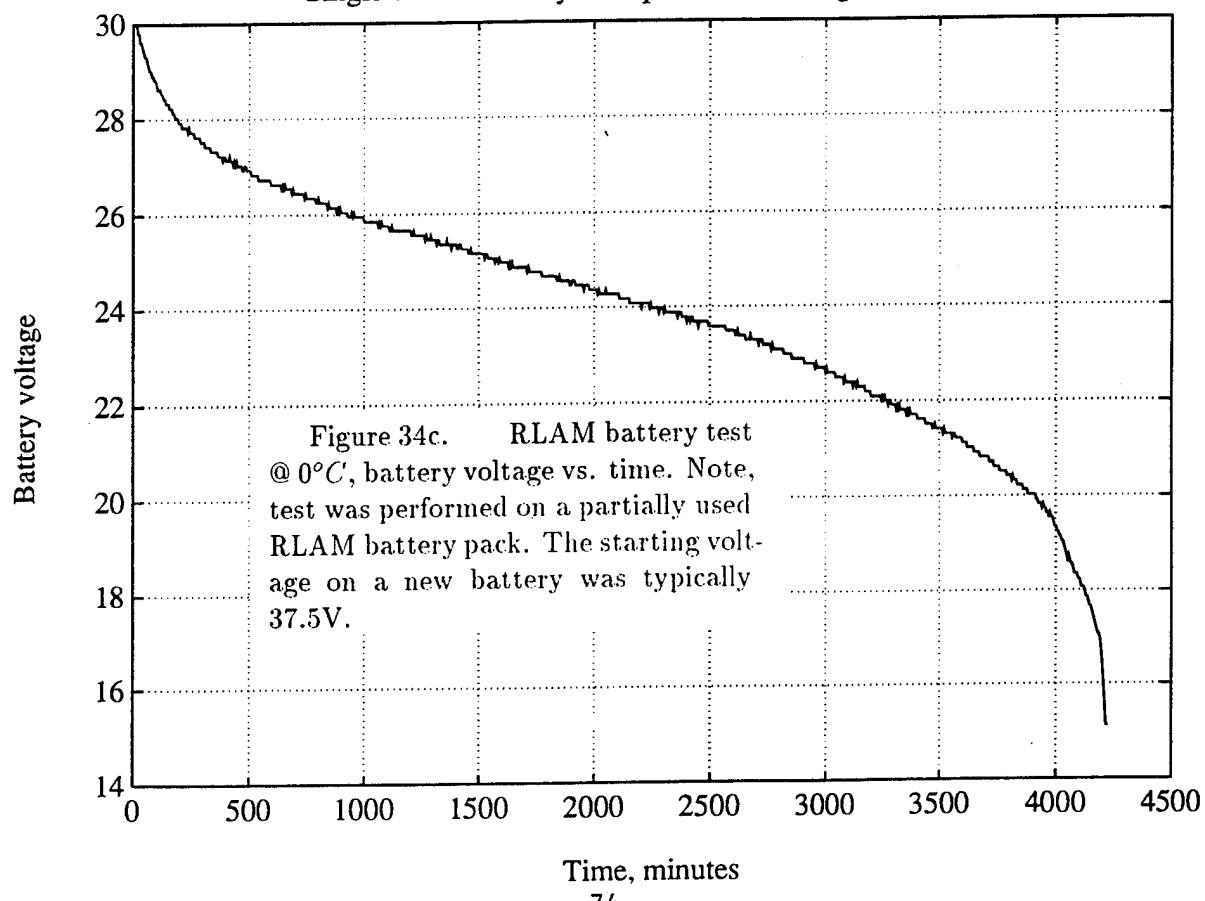
Single RLAM battery test, #3, 0 degrees Celsius

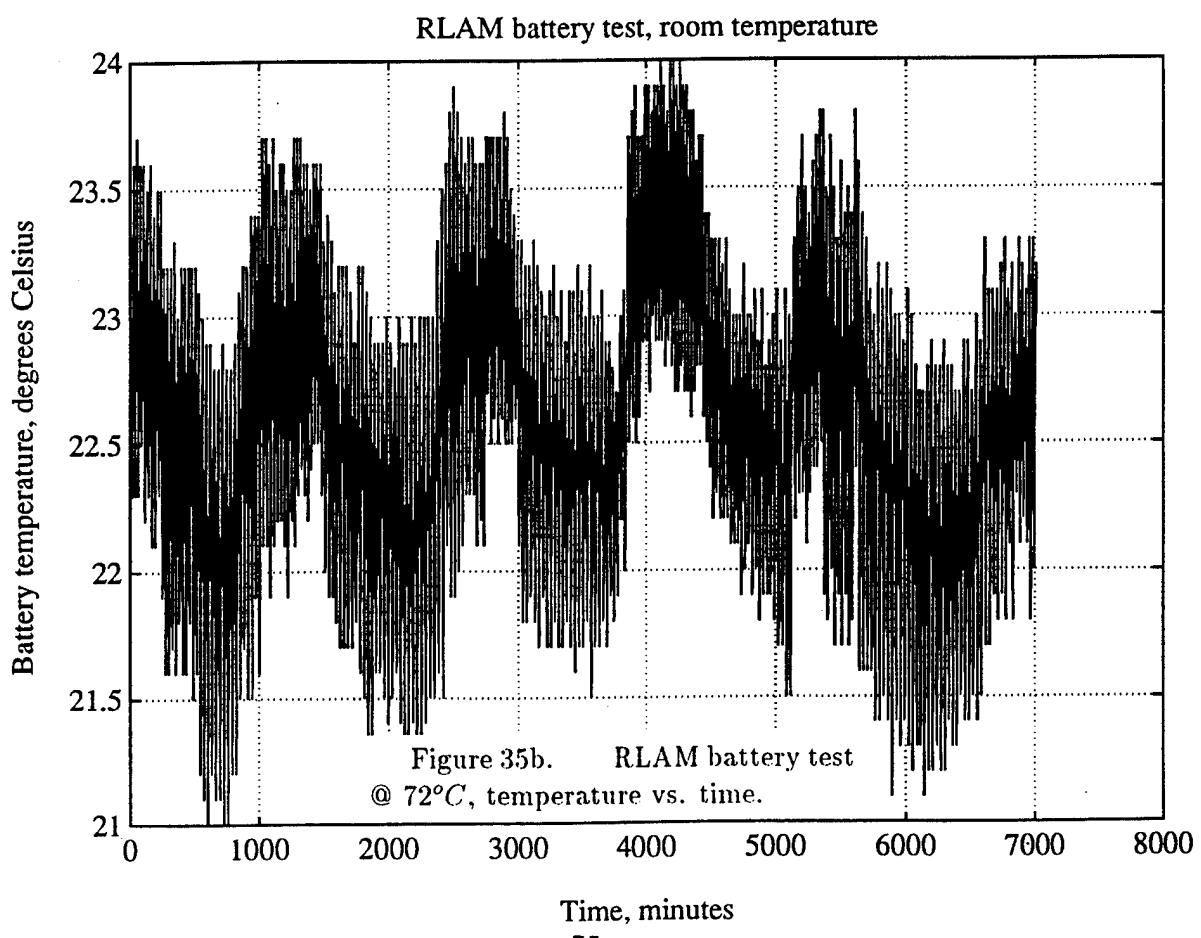
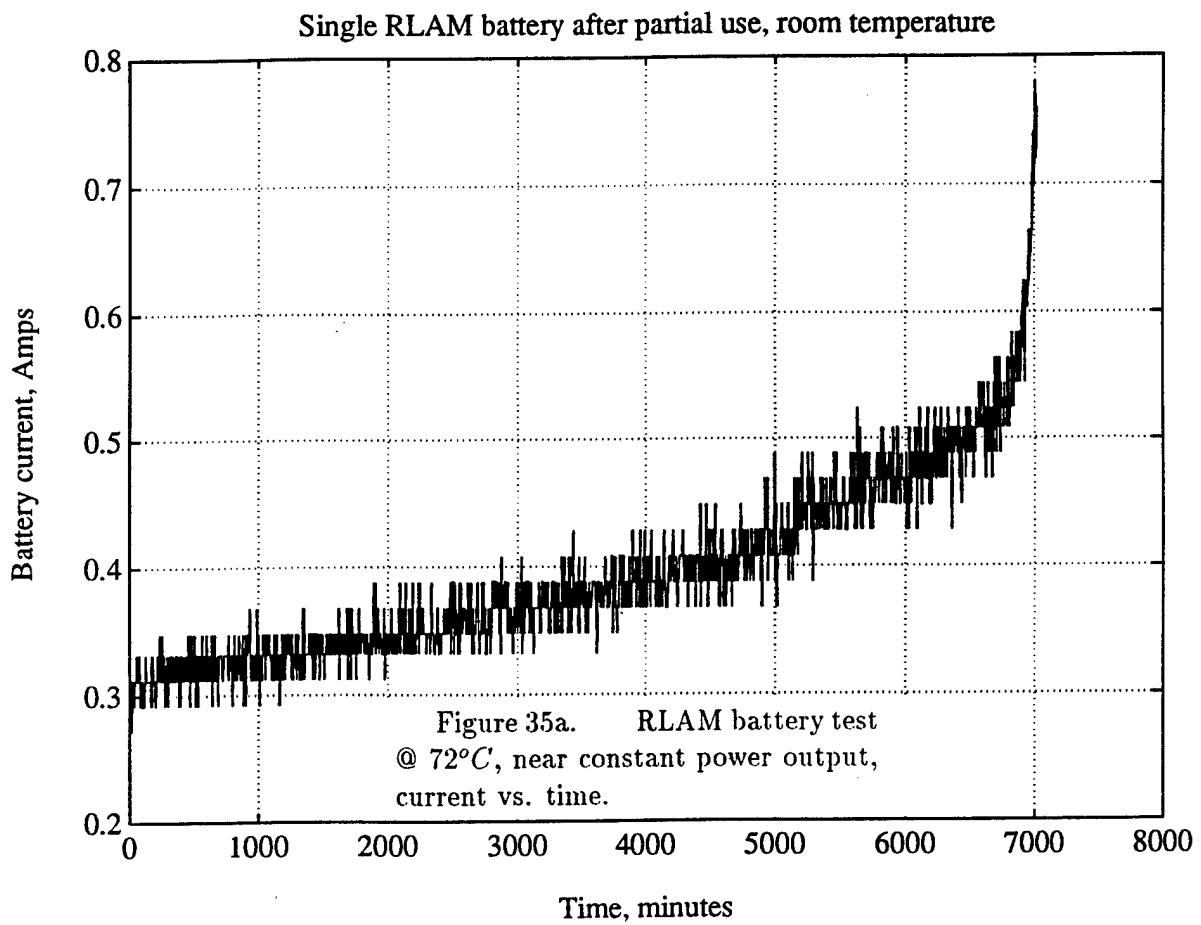


Single RLAM battery after partial use, 0 degrees Celcius

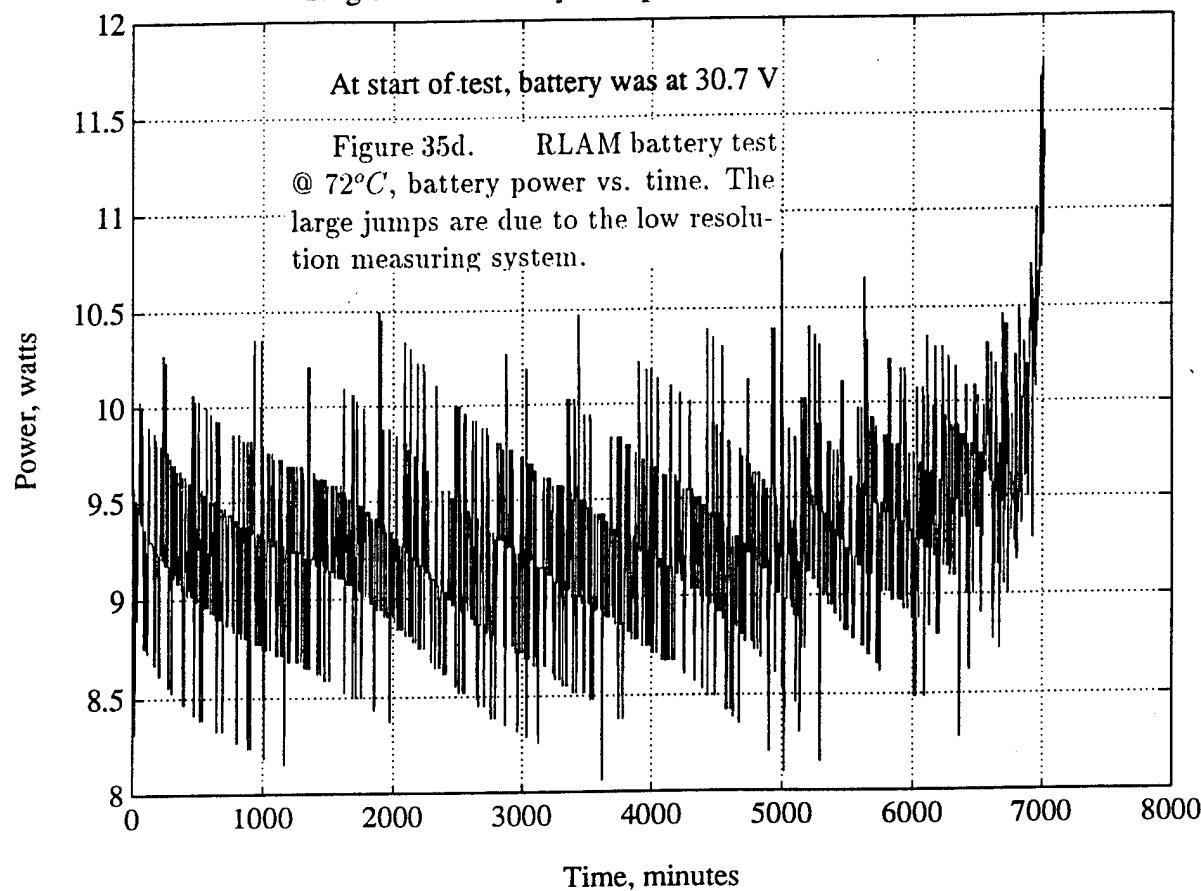


Single RLAM battery after partial use, 0 degrees Celcius





Single RLAM battery after partial use, room temperature



Single RLAM battery after partial use, room temperature

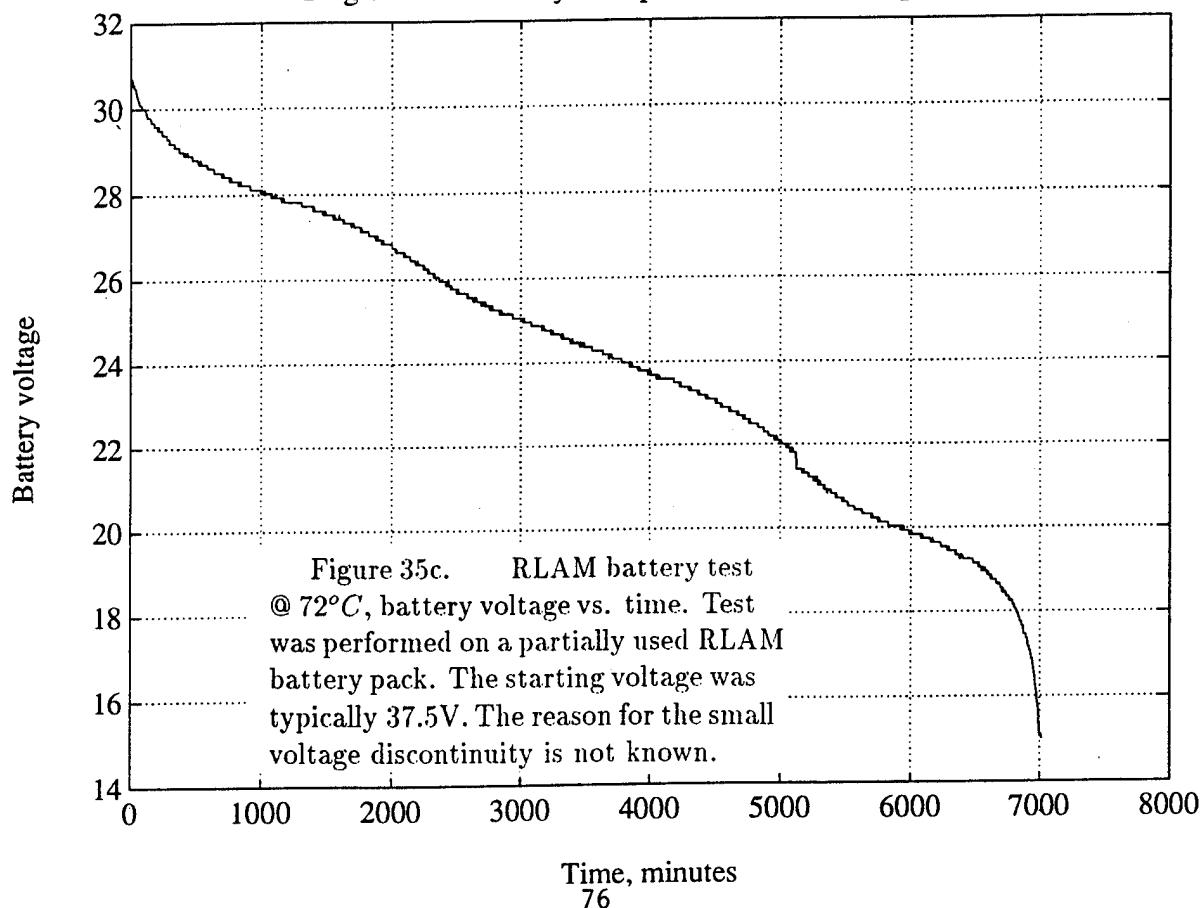


TABLE 1  
24 BIT A/D AQUISITION CARD JUMPER LIST

JUMPER #	SOLDER/ PUSH-ON	DEFAULT CLOSED	FUNCTION
JP1A-H	S	X	AD620 GAIN = 20 DB
JP2A-H	S		AD620 GAIN = 40 DB
			(NEITHER JUMPER = 0 DB)
JP3A-H	S		SENSOR VAMP+ FILTER BYPASS
JP4A-H	S	X	POSITIVE SIDE CURRENT MODE LOAD
JP5A-H	S		VOLTAGE MODE LOAD CONNECT
JP6A-H	S	X	NEGATIVE SIDE CURRENT MODE LOAD
JP7A-H	S	X	EACH CHANNEL ANALOG GND TO A/D FILTER GND
JP8	S		COMMON ANALOG GND TO A/D FILTER GND
JP8A-H	S	X	PREAMPLIFIER OUTPUTS CONNECTION TO A/D
JP9	PO	X	PC BUS +5V TO CARD LOGIC
JP10	PO	X	PC BUS -12V TO CARD REGULATORS
JP11	PO	X	PC BUS +12V TO CARD REGULATORS
JP12			FIFO HALF FULL FLAG INTERRUPT
1-2	PO		IRQB - IRQ3
3-4	PO		IRQB - IRQ4
5-6	PO		IRQB - IRQ7
7-8	PO		IRQB - IRQ10
9-10	PO	X	IRQB - IRQ11
11-12	PO		IRQB - IRQ12
13-14	PO		IRQB - IRQ15
JP19			MODULO n FRAME TIME MARK INTERRUPT
1-2	PO		IRQA - IRQ3
3-4	PO		IRQA - IRQ4
5-6	PO		IRQA - IRQ7
7-8	PO	X	IRQA - IRQ10
9-10	PO		IRQA - IRQ11
11-12	PO		IRQA - IRQ12
13-14	PO		IRQA - IRQ15
JP26	PO		EXTERNAL SUPPLY +8V TO VAMP+
JP27	PO	X	INTERNAL +8V REGULATOR TO VAMP+
JP28	PO		EXTERNAL SUPPLY COMMON TO ANALOG GND
JP29	PO	X	LOGIC DIGITAL GND TO ANALOG GND
JP30	PO	X	EXTERNAL SUPPLY -8V TO VAMP-
JP31	PO	X	INTERNAL -8V REGULATOR TO VAMP-
JP32	PO		EXTERNAL SUPPLY +8V TO A/D CHIP +5V REG
JP33	PO	X	PC BUS +12V THRU JP11 TO A/D CHIP +5V REG
JP34	PO		EXTERNAL SUPPLY -8V TO A/D CHIP -5V REG
JP35	PO	X	PC BUS -12V THRU JP10 TO A/D CHIP -5V REG
JP36	PO		EXTERNAL SUPPLY +5V TO LOGIC
JP37	PO		EXTERNAL SUPPLY +5V TO A/D FILTER +5V
JP38	PO	X	PC BUS +5V THRU JP9 TO A/D FILTER +5V
JP40	S	X	EXTERNAL SUPPLY +5V RTN TO LOGIC GND
JP41	S		LOGIC GND TO A/D FILTER DIGITAL GND
JP42	S		SUITE COUNTER DIVIDE BY 128 OUTPUT
JP43	S		" " " " 256 "
JP44	S	X	" " " " 512 "
JP45	S		" " " " 1024 "
JP46	S		" " " " 2048 "
JP47	S		" " " " 4096 "
JP48	S	X	ON-BOARD 4MHZ OSCILLATOR DISABLE
JP49	S		ON-BOARD 1MHZ CLOCK CONNECT
JP50	S		ON-BOARD 2MHZ CLOCK CONNECT
JP51	S		ON-BOARD CLOCK BUFFER ENABLE
JP52	S	X	1MHZ CLOCK TO LOGIC BUFFER CONNECT
JP53	S		ON-BOARD 1MHZ CLOCK TO LOGIC BUFFER CONNECT
JP54	S	X	CLOCK FREQUENCY DOUBLER ENABLE

Table 2, 1024 byte Data Record Header (DRH), as on tape

000000	44 41 54 41 53 49 4d 49 20 52 4c 41 4d 00 00 00	DATASIMI RLAM...
000010	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000020	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000030	00 00 00 00 00 00 00 00 00 00 38 04 a8 61 10 00	.....8.a..
000040	00 00 00 2a 00 00 00 24 f4 44 f1 03 00 00 00 00	....*....\$D....
000050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000060	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000070	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000080	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000090	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000a0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000b0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000c0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000d0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000e0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0000f0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000100	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000110	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000120	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000130	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000140	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000150	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000160	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000170	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000180	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000190	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0001a0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0001b0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0001c0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0001d0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0001e0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4c 54	.....LT
0001f0	58 32 00 00 ca 07 70 00 38 04 a8 61 54 00 00 00	X2....p.8..aT...
000200	00 00 00 01 00 15 05 10 00 00 10 00 00 00 4c 54	.....LT
000210	58 33 00 00 ca 07 70 00 38 04 a8 61 51 00 00 00	X3....p.8..aQ...
000220	00 00 00 01 00 15 05 10 00 00 10 00 00 00 4c 54	.....LT
000230	58 35 00 00 ca 07 70 00 38 04 a8 61 51 00 00 00	X5....p.8..aQ...
000240	00 00 00 01 00 15 05 10 00 00 10 00 00 00 00 00	.....
000250	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000260	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000270	54 00 02 00 64 00 00 02 00 00 00 00 00 00 00 01	T...d.....
000280	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000290	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002a0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002b0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002c0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002d0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002e0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0002f0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000300	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000310	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000320	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000330	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000340	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000350	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000360	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000370	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000380	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000390	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0003a0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0003b0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0003c0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0003d0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
0003e0	52 30 00 00 00 00 00 00 00 00 30 34 32 33	R0.....0423
0003f0	30 35 35 38 2e 44 41 54 00 00 00 44 41 54 41	0558.DAT....DATA

## DOCUMENT LIBRARY

*Distribution List for Technical Report Exchange - May 1995*

University of California, San Diego  
SIO Library 0175C  
9500 Gilman Drive  
La Jolla, CA 92093-0175

Hancock Library of Biology & Oceanography  
Alan Hancock Laboratory  
University of Southern California  
University Park  
Los Angeles, CA 90089-0371

Gifts & Exchanges  
Library  
Bedford Institute of Oceanography  
P.O. Box 1006  
Dartmouth, NS, B2Y 4A2, CANADA

Commander  
International Ice Patrol  
1082 Shennecossett Road  
Groton, CT 06340-6095

NOAA/EDIS Miami Library Center  
4301 Rickenbacker Causeway  
Miami, FL 33149

Research Library  
U.S. Army Corps of Engineers  
Waterways Experiment Station  
3909 Halls Ferry Road  
Vicksburg, MS 39180-6199

Institute of Geophysics  
University of Hawaii  
Library Room 252  
2525 Correa Road  
Honolulu, HI 96822

Marine Resources Information Center  
Building E38-320  
MIT  
Cambridge, MA 02139

Library  
Lamont-Doherty Geological Observatory  
Columbia University  
Palisades, NY z10964

Library  
Serials Department  
Oregon State University  
Corvallis, OR 97331

Pell Marine Science Library  
University of Rhode Island  
Narragansett Bay Campus  
Narragansett, RI 02882

Working Collection  
Texas A&M University  
Dept. of Oceanography  
College Station, TX 77843

Fisheries-Oceanography Library  
151 Oceanography Teaching Bldg.  
University of Washington  
Seattle, WA 98195

Library  
R.S.M.A.S.  
University of Miami  
4600 Rickenbacker Causeway  
Miami, FL 33149

Maury Oceanographic Library  
Naval Oceanographic Office  
Building 1003 South  
1002 Balch Blvd.  
Stennis Space Center, MS, 39522-5001

Library  
Institute of Ocean Sciences  
P.O. Box 6000  
Sidney, B.C. V8L 4B2  
CANADA

Library  
Institute of Oceanographic Sciences  
Deacon Laboratory  
Wormley, Godalming  
Surrey GU8 5UB  
UNITED KINGDOM

The Librarian  
CSIRO Marine Laboratories  
G.P.O. Box 1538  
Hobart, Tasmania  
AUSTRALIA 7001

Library  
Proudman Oceanographic Laboratory  
Bidston Observatory  
Birkenhead  
Merseyside L43 7 RA  
UNITED KINGDOM

IFREMER  
Centre de Brest  
Service Documentation - Publications  
BP 70 29280 PLOUZANE  
FRANCE

<b>REPORT DOCUMENTATION PAGE</b>		<b>1. REPORT NO.</b> <b>WHOI-95-09</b>	<b>2</b>	<b>3. Recipient's Accession No.</b>
<b>4. Title and Subtitle</b> Radio LAN Acquisition Module (RLAM), Recent Developments for High Resolution Data Collection Systems as Implemented for the ONR Sea Ice Mechanics Experiment, Spring 1994		<b>5. Report Date</b> May 1995		
<b>7. Author(s)</b> K. von der Heydt, C.F. Eck		<b>8. Performing Organization Rept. No.</b> WHOI-95-09		
<b>9. Performing Organization Name and Address</b>  Woods Hole Oceanographic Institution Woods Hole, Massachusetts 02543		<b>10. Project/Task/Work Unit No.</b>		
		<b>11. Contract(C) or Grant(G) No.</b> (C) N00014-91-J-1296 (G)		
<b>12. Sponsoring Organization Name and Address</b>  Office of Naval Research		<b>13. Type of Report &amp; Period Covered</b> Technical Report		
		<b>14.</b>		
<b>15. Supplementary Notes</b> This report should be cited as: Woods Hole Oceanog. Inst. Tech. Rept., WHOI-95-09.				
<b>16. Abstract (Limit: 200 words)</b>  During a recent experiment (April 1994), for the ONR Sea Ice Mechanics Initiative (SIMI), a portable data acquisition system was assembled that included 2 new developments. The first consists of a board, designed for the ISA PC bus incorporating 8 - 24 bit sigma-delta analog-to-digital converter (ADC) channels with 20 bit rms dynamic range. Among the features are programmable bandwidth to 1500 Hz, low power dissipation, digital anti-alias filetering, and a "floating point" mode resulting in a 16 bit word. Secondly, since the telemetry of data at continuous rates in excess of 100Kbytes/s was required, hardware & software was developed to use a wireless LAN to network 3 sites up to 5km distant from the data recording system. Details of the system along with test data are described.				
<b>17. Document Analysis</b> <b>a. Descriptors</b> data acquisition systems local area networks sea ice mechanics				
<b>b. Identifiers/Open-Ended Terms</b>				
<b>c. COSATI Field/Group</b>				
<b>18. Availability Statement</b>  Approved for public release; distribution unlimited.		<b>19. Security Class (This Report)</b> <b>UNCLASSIFIED</b>	<b>21. No. of Pages</b> 82	
		<b>20. Security Class (This Page)</b>	<b>22. Price</b>	